

1. TITLE OF THE INVENTION

A Multiplexing Frame Modification Method, A Synchronization Flag Modification Method, And Multiplexing Header Extraction Method

5 [CLAIMS]

1 A multiplexing frame modification method which a frame length of a multiplexing frame having a synchronization flag is semi-fixed, and sound data, video data, other data or any combinations of sound data, video data or other data are multiplexed and transmitted; and when a transmission side
10 modifies said semi-fixed frame length from $F \times N$ byte (F and N are a natural number) to $F \times M$ byte (M is a natural number excluding N), a reception side performs synchronization flag extraction process per F byte.

2. A multiplexing frame modification method which a frame length of a multiplexing frame having a synchronization flag is semi-fixed, and sound
15 data, video data, other data or any combinations of sound data, video data or other data are multiplexed and transmitted, and when a transmission side modifies said semi-fixed frame length from $F \times N$ byte (F and N are a natural number) to $F \times M$ byte (M is a natural number excluding N), a
20 reception side performs synchronization flag extraction process per $F \times L$ byte (L is the greatest common divisor of N and M).

3. A multiplexing frame modification method according to Claim 1 and 2,
wherein a transmission side a modification request of multiplexing frame
25 length, and a reception side returns a confirmation response for the modification request, and when a transmission side modifies multiplexing frame length after receiving said confirmation response, a reception side performs synchronization flag extraction process per F byte or $F \times L$ byte (L is the greatest common divisor of N and M) after returning said

confirmation response.

4. A multiplexing frame modification method according to Claim 1 and 2,
wherein a transmission side a modification request of multiplexing frame
5 length, and a reception side returns a confirmation response for the
modification request; and a transmission side measures response delay time
for responding said confirmation response in advance, and notifies a
reception side of the measurement result of said response delay time; and
when a transmission side modifies multiplexing frame length after
10 receiving said confirmation response, a reception side performs
synchronization flag extraction process per F byte or F x L byte after
passing said response delay time at the time of returning a confirmation
response.

15 5. A multiplexing frame modification method according to one of Claims
1 to 4,

wherein, as a result of synchronization flag extraction process performed
per F byte or F x L byte, when a synchronization flag more than
predetermined times is extracted in succession per F x M byte,
20 synchronization flag extraction process is performed in the reception side
per F x M byte.

6. A multiplexing frame modification method which a frame length of a
multiplexing frame is semi-fixed after frame synchronization is established
25 by using a synchronization frame of F x N byte, and sound data, video data,
other data or any combinations of sound data, video data or other data are
multiplexed and transmitted by using synchronization in the digital network
and a multiplexing frame without a synchronization flag; and when a
transmission side modifies said semi-fixed frame length from F x N byte (F

and N are a natural number) to $F \times M$ byte (M is a natural number excluding N), a reception side performs synchronization flag extraction process per F byte.

- 5 7. A multiplexing frame modification method which a frame length of a multiplexing frame is semi-fixed after frame synchronization is established by using a synchronization frame of $F \times N$ byte, and sound data, video data, other data or any combinations of sound data, video data or other data are multiplexed and transmitted by using synchronization in the digital network
- 10 and a multiplexing frame without a synchronization flag; and when a transmission side modifies said semi-fixed frame length from $F \times N$ byte (F and N are a natural number) to $F \times M$ byte (M is a natural number excluding N), a reception side performs synchronization flag extraction process per $F \times L$ byte (L is the greatest common divisor of N and M).
- 15
8. A multiplexing frame modification method according to Claim 6 and 7, wherein a transmission side a modification request of multiplexing frame length, and a reception side returns a confirmation response for the modification request, and when a transmission side modifies multiplexing
- 20 frame length after receiving said confirmation response, a reception side performs multiplexing header extraction process per F byte or $F \times L$ byte (L is the greatest common divisor of N and M) after returning said confirmation response.
- 25 9. A multiplexing frame modification method according to Claim 6 and 7, wherein a transmission side a modification request of multiplexing frame length, and a reception side returns a confirmation response for the modification request; and a transmission side measures response delay time for responding said confirmation response in advance, and notifies a

reception side of the measurement result of said response delay time; and when a transmission side modifies multiplexing frame length after receiving said confirmation response, a reception side performs multiplexing header extraction process per F byte or F x L byte after
5 passing said response delay time at the time of returning a confirmation response.

10. A multiplexing frame modification method according to one of Claims 6 to 9,

10 wherein the field is transmitted to a higher layer as information filed, the field between multiplexing headers extracted by said multiplexing header extraction process.

11 A multiplexing frame modification method according to one of Claims
15 6 to 10,

wherein it is modified not to interleave a multiplexing header before transmitting a modification command of said multiplexing header frame length.

20 12. A multiplexing frame modification method according to one of Claims 1 to 5,

wherein pattern matching is performed as synchronization flag extraction process, the pattern matching for comparing synchronization pattern of predetermined numbers of byte to received data by shifting the received
25 data by one byte.

13. A multiplexing frame modification method according to one of Claims 6 to 11,

wherein error detection of a multiplexing information unit is performed

by using an error detection code as multiplexing header extraction process after an error correction of a multiplexing information unit and an error correction code is performed by using an error correction code attached to a multiplexing header, and if no error is detected, it is determined as a multiplexing header.

14. A multiplexing frame modification method according to one of Claims 2 to 4, or one of Claims 7 to 12,

wherein said method is either $F=20$, $N=4$, $M=2$, or $L=2$.

15. A multiplexing frame modification method according to one of Claims 2 to 4, or one of Claims 7 to 12,

wherein said method is either $F=20$, $N=4$, $M=3$, or $L=1$.

16. A multiplexing frame modification method according to one of Claims 2 to 4, or one of Claims 7 to 12,

wherein said method is either $F=20$, $N=4$, $M=6$, or $L=2$.

17. A multiplexing frame modification method according to one of Claims 1 to 6,

wherein said method is either $F=20$, $N=4$, or $M=2$.

18. A multiplexing frame modification method according to one of Claims 1 to 6,

wherein said method is either $F=20$, $N=4$, or $M=3$.

19. A multiplexing frame modification method according to one of Claims 1 to 6,

wherein said method is either $F=20$, $N=4$, or $M=6$

20. A multiplexing frame modification method which a frame length of a multiplexing frame is semi-fixed after frame synchronization is established by using a synchronization frame of $F \times N$ byte (F and N are a natural number), and sound data, video data, other data or any combinations of sound data, video data or other data are multiplexed and transmitted by using synchronization in the digital network and a multiplexing frame without a synchronization flag; and multiplexing header extraction process is performed within a range of on or about P byte (P is an integer) of the reference position by determining the reference position of next multiplexing frame after $F \times N$ byte from the reference position of a multiplexing header of last multiplexing frame.

21. A multiplexing frame modification method according to Claim 20, wherein multiplexing header extraction process is firstly performed, multiplexing header extraction process is secondly performed within a range which is shifted on or about J byte from the reference position, and multiplexing header extraction process is performed J byte in 1 byte steps until a range of shifting on or about P byte (P is an integer) from the reference position.

22. A multiplexing frame modification method according to Claim 21, wherein, as a result of multiplexing header extraction process, when a multiplexing header is correctly extracted, multiplexing header extraction process is stopped in the reference position where said multiplexing header is correctly extracted afterwards.

23. A multiplexing frame modification method according to one of Claims 20 to 22,

wherein a multiplexing separation is performed by determining $F \times N$ byte which after headers are extracted as information field.

24. A multiplexing frame modification method according to one of
5 Claims 20 to 23,

wherein when a position of extracted multiplexing headers is shifted from a reference position for p bytes(P is an integer) S times(S is a natural number), the reference position for the next multiplexing frame is shifted for p bytes.

10 25. A multiplexing frame modification method according to one of Claims 20 to 24,

wherein when a multiplexing header is not extracted $E1$ time ($E1$ is a natural number) in succession from a range of $\pm P$ bytes of the reference point, it is assumed that synchronization shifts occur and the process is
15 performed for establishing frame synchronization by using a synchronization frame of $F \times N$ bytes(F, N are a natural number) having a synchronization flag.

20 26. A multiplexing frame modification method according to one of Claims 20 to 24,

wherein only a multiplexing header is performed at the reference position as $P=0$ in a normal condition, and as a result, when a multiplexing header is not extracted $E2$ times(E is a natural number) in succession, it is
25 set as $P \geq 1$, consequently, when the extracted position of a multiplexing header is shifted from the reference position for p bytes(P is an integer) S times(S is a natural number), the reference position for the next multiplexing frame is shifted for p bytes, and multiplexing header extraction process is moved to the normal condition as $P=0$.

27. A multiplexing frame modification method when a multiplexing frame of $F \times N$ byte (F, N are a natural number) having a synchronization flag is modified to a multiplexing frame of $F \times N$ byte without a synchronization flag, a frame having a synchronization flag is received and multiplexing header extraction process is performed per $F \times N$ byte in parallel.
28. A multiplexing frame modification method when a multiplexing frame of $F \times N$ byte (F, N are a natural number) having a synchronization flag is modified to a multiplexing frame of $F \times M$ byte (M is a natural number excluding N) without a synchronization flag, a frame having a synchronization flag is received and multiplexing header extraction process is performed per $F \times N$ byte in parallel, and multiplexing header extraction process is performed per F byte or $F \times L$ byte (L is the greatest common divisor of N and M).
29. A multiplexing frame modification method according to Claim 27 or 28,
- wherein when a transmission side transmits a synchronization flag modification command, a reception side returns a confirmation response for said synchronization flag modification command, and when a transmission side modifies a synchronization flag after receiving said confirmation response, a reception side performs multiplexing header extraction process per $F \times N$ byte or per F byte or per $F \times L$ byte after returning a confirmation response.
30. A multiplexing frame modification method according to one of Claims 27 to 30,

wherein when a transmission side transmits a synchronization flag modification command, a reception side returns a confirmation response for said synchronization flag modification command, and when a transmission side measures response delay time required for said confirmation response in advance, a transmission side notifies a reception side of a measurement result of said response delay time, and when a transmission side modifies a synchronization flag after receiving said confirmation response, a reception side performs multiplex header extraction process per $F \times N$ byte or per F byte or per $F \times L$ byte header at the time after said response delay time is over from the time that said response delay time is returned

31. A multiplexing frame modification method according to one of Claims 27 to 30,

wherein a multiplexing frame is received in parallel, the multiplexing frame having a synchronization flag and without a synchronization flag until a multiplexing header is extracted S times in succession.

32. A multiplexing frame modification method according to one of Claims 27 to 31,

wherein when error correction capability of a multiplexing header, or error detection capability of a multiplexing header is modified, or both capabilities of a multiplexing header are modified at the same time of modifying a synchronization flag, error correction and error detection of a multiplexing header are performed by using error correction capability and error detection capability both of which are not modified with regard to receiving a multiplexing frame having said synchronization flag, and a multiplexing header extraction process is performed by using error correction capability and error detection capability both of which are

modified with regard to multiplexing header extraction process performed at the front of said multiplexing frame.

33. A multiplexing frame modification method according to Claim 31 or 32,

wherein parameter modification except for a parameter of said synchronization flag and said multiplexing header is performed from when a multiplexing header is extracted at the front of said multiplexing frame S time in succession.

[Embodiment 1]

Figure 1 is a timing figure according to the embodiment 1 when a frame length of a multiplexing frame, having a synchronization flag at the head, is modified from 80 bytes to 60 bytes. A radio slot of a PHS is 20 bytes, and $F = 20$ is suitable (F : natural number). In fig.1, (a) is a timing figure after one frame when multiplexing frame length is modified to 60 bytes. (b) is a timing figure after two frames when multiplexing frame length is modified to 60 bytes. (c) is a timing figure after three frames when multiplexing frame length is modified to 60 bytes. (d) is a timing figure after four frames when multiplexing frame length is modified to 60 bytes. (e) is a timing figure of the multiplexing flag extraction process when multiplexing frame length is modified according to the embodiment 1. In fig.2, (a) is a timing figure when multiplexing frame length is modified to 40 bytes after one frame. (b) is a timing figure when multiplexing frame length is modified to 40 bytes after two frames. (c) is a timing figure when multiplexing frame length is modified to 40 bytes after three frames. (d) is a timing figure when multiplexing frame length is modified to 40 bytes after four frames. (e) is a timing figure of the multiplexing flag extraction process when multiplexing frame length is modified according

to the embodiment 1. In fig.1, a multiplexing frame is a multiplexing frame defined in H223/Annex A described in the prior art in fig.9, the description of this multiplexing frame will be omitted in this paragraph. The filed written as synchronization is a synchronization flag , and a header and an information field are stored in the other fields. The operation will be described by using the above timing figures and figure 3.

In fig.1, a hatched synchronization flag is a synchronization flag of a multiplexing frame whose the frame length is 60 bytes. Fig.1(a) to fig.1(d) show that a timing is shifted one frame, the timing which a frame length is switched from 80 bytes to 60 bytes. With regard to the hatched synchronization flags, a synchronization flag in (b) is delayed for 20 bytes in comparison with a synchronization flag in (a), and a synchronization flag in (c) is delayed for 20 bytes in comparison with a synchronization flag in (b). Further, a synchronization flag in (d) is delayed for 20 bytes in comparison with a synchronization flag in (c), however, the timing of a synchronization flag in (d) is the same as that of a synchronization flag in (a). That is to say, a synchronization flag of 60 bytes is placed in one of three phases each of which has 20 bytes, and the timing of a synchronization flag of 80 bytes, prior to the above synchronization flag of 60 bytes, is the same as that of one of three synchronization flags(60 bytes) from (a) to (c). Therefore, it is understood that a synchronization flag extraction process is minimally necessary per 20 byte when a frame length of a multiplexing frame is modified from 80 bytes to 60 bytes. A synchronization flag extraction process is to compare synchronization pattern of predetermined bytes with received data by shifting the received data, and to perform pattern matching. Further, in a synchronization flag extraction process, a synchronization flag of 4 bytes [1111100011011101010000100101100] is used in the initial state as the synchronization pattern.

In H.223/AnnexA, decreasing of the process and lowering of extraction by mistake are implemented by that a synchronization flag extraction process is only performed to on or about bytes of a fixed frame length per a fixed frame length by that a frame length of a multiplexing frame is semi-fixed. On or about bytes is selected within a range of absorbing a synchronization slip in the network, the synchronization slip caused by an error of clock precision of an exchange.

As shown in fig.2, it is described that a 80 byte frame length is modified to 40 byte. A hatched synchronization flag of 40 bytes is placed per 40 bytes, and a synchronization flag of 80 bytes, prior to a hatched synchronization flag of 40 bytes, is placed per 80 bytes. That is to say, it is necessary that a synchronization flag process is performed per 40 bytes. Extraction of the synchronization may be performed in the case of performing a synchronization flag extraction process per 20 bytes, however, the throughput is less necessary in the case of 40 bytes. For the above reason, when a frame length is modified from a multiplexing frame of $20 \times N$ byte (N : natural number) to a multiplexing frame of $20 \times M$ byte (M : natural number excluding N), a synchronization flag extraction process is performed per 20 bytes, or a synchronization flag extraction process is performed per $20 \times L$ bytes (L is the greatest common divisor of N and M). Further, with regard to a command and a sequence both of which modify a frame length of a multiplexing frame defined in H.223/AnnexA described in fig.15, the command and sequence are performed after a confirmation response for a frame length modification command is transmitted in the receiving end, and the synchronization flag extraction process, for a fixed length frame of a frame length after the modification, is performed at the time of extracting a synchronization flag prescribed times, the synchronization flag after the change of a frame length. A flow chart for the above operation will be described in fig.3. In fig.3, 301 indicates the state

before multiplexing frame length is modified, and a synchronization flag extraction process is performed by $20 \times N$ byte cycle. The number in a parenthesis is one of the examples. The state is repeated until a confirmation response of frame length modification is transmitted in the receiving end. In 302, the process goes to a mode for performing a synchronization flag extraction process by $20 \times L$ byte in the case of Yes. In 303, variable s is initialized, the variable s for protecting synchronization. In 304, variable m is initialized, m which extracts a synchronization flag interval of $20 \times M$ byte. In 305, a synchronization flag extraction process is performed after $20 \times L$ byte. In 306, if no synchronization flag exists, m is incremented in 307. In 306, when a synchronization flag is extracted, the field placed in between the last extracted synchronization flag and the extracted synchronization flag is extracted as a header and an information filed in 308, and a header and an information filed are transmitted to a higher layer. In 309, it is determined whether the interval between synchronization flags is $20 \times M$ byte or not. And, when the interval between synchronization flags is not $20 \times M$ byte, variable s is initialized, the variable s for protecting synchronization, and variable m is initialized, the variable m for extracting a synchronization flag interval in 304, a synchronization flag extraction process of $20 \times L$ byte continues in 305. In 309, when a synchronization flag interval is $20 \times M$ byte, a synchronization flag interval is checked for protecting synchronization in 310. S is the predetermined extraction number of successive synchronization flag for protecting synchronization, and when s is less than S , s is incremented in 312, variable m is initialized, the variable m for extracting a synchronization flag interval in 304, and a synchronization flag extraction process of $20 \times L$ byte continues in 305. If $s=S$ in 310, it is determined that synchronization of a frame length is established, the frame length after being modified, and the process goes to reception 313 of $20 \times M$ byte

cycle.

According to the embodiment 1, a frame length is modified promptly by small throughput in the case of modifying a frame length of a semi-fixed multiplexing frame having a synchronization flag.

5 If the time for responding delay time is understood(e.g. MODS ARQ), the delay time from which a confirmation response is transmitted to which a frame length after the modification is transmitted, the throughput is decreased when a synchronization flag process is modified to the process per 20 byte, or per $20 \times L$ byte after the response delay time from the time
10 of transmitting a response delay time. Further, the case in $F=20$ is described in this embodiment, however, the invention has the same effect if F is other values, and if N and M are other values not described in the above case.

In this case, offset is shifted in $J=1$ steps, the offset for multiplexing header extraction process, however, when a synchronization shift of the
15 network is occurred per 4 bytes, multiplexing header extraction process should be performed in $J=4$ steps(=4 bytes).

[Embodiment 2]

Fig.4 is a timing figure when a frame length of a multiplexing frame is modified from 80 bytes to 60 bytes according to the embodiment 2 of the
20 present invention, the frame length without a synchronization flag at the front. A radio slot of a PHS is 20 bytes, and $F=20$ is suitable. In fig.4, fig.(a) is a timing figure after one frame when multiplexing frame length is modified to 60 bytes, (b) is a timing figure after two frames when multiplexing frame length is modified to 60 bytes, (c) is a timing figure
25 after three frames when multiplexing frame length is modified to 60 bytes, (d) is a timing figure after four frames when multiplexing frame length is modified to 60 bytes, and (e) is a timing figure of the multiplexing flag extraction process when multiplexing frame length is modified according to the embodiment 2. In fig.4, a multiplexing frame is a multiplexing frame

which extends H.223/Annex A described in fig.12, and deletes a synchronization flag, and the description of this multiplexing frame will be described. The field written as MC is a multiplexing header described in fig.10(prior art), an information field is stored in the other fields. The
5 operation will be described by using a flow chart in fig.4 and fig.5.

In fig.4, a hatched multiplexing frame is a multiplexing header of a multiplexing frame of a frame length of 60 bytes. Fig.1(a) to fig.1(d) show that a timing is shifted one frame, the timing which a frame length is switched from 80 bytes to 60 bytes.

10 With regard to the hatched multiplexing headers, a multiplexing header in (b) is delayed for 20 bytes in comparison with a multiplexing header in (a), and a multiplexing header in (c) is delayed for 20 bytes in comparison with a multiplexing header in (b). Further, a multiplexing header in (d) is delayed for 20 bytes in comparison with a multiplexing header in (c),
15 however, the timing of a multiplexing header in (d) is the same as that of a multiplexing header in (a). That is to say, a multiplexing header of 60 bytes is placed in one of three phases each of which has 20 bytes, and the timing of a multiplexing header of 80 bytes, prior to the above multiplexing header of 60 bytes, is the same as that of one of three multiplexing headers(60
20 bytes) from (a) to (c). Therefore, it is understood that a synchronization flag extraction process is minimally necessary per 20 byte when a frame length of a multiplexing frame is modified from 80 bytes to 60 bytes. A multiplexing header extraction process is to make a correction to a multiplexing information unit and an error detection code by using an error
25 correction code attached to a multiplexing header, and to detect an error of a multiplexing information unit by using an error detection code, and when an error is not detected, a multiplexing header is extracted. Further, in the multiplexing header extraction process, a search on or about bytes may be performed or not within a range of absorbing a bit slip in the network, the

bit slip caused by an error of clock precision of an exchange.

The result in fig.4 is similar to that of fig.1 of the embodiment 1, and when a frame length is modified from 80 bytes to 20 bytes, the result is the same as that of fig.2, therefore, the description will be omitted. When a
 5 multiplexing frame of $20 \times N$ byte is modified to a multiplexing frame of $20 \times M$ byte (M : natural number excluding N), a synchronization flag extraction process is performed per 20 bytes, or a synchronization flag extraction process is performed per $20 \times L$ bytes (L is the greatest common
 10 divisor of N and M). Further, with regard to a command and a sequence both of which modify a frame length of a multiplexing frame defined in H.223/AnnexA described in fig.15, the command and sequence are performed after a confirmation response for a frame length modification
 command is transmitted in the receiving end, and the synchronization flag extraction process, for a fixed length frame of a frame length after the
 15 modification, is performed at the time of extracting a multiplexing frame prescribed times, the multiplexing frame after the change of a frame length. A flow chart for the above operation will be described in fig 5. In fig.5, 501 indicates the state before multiplexing frame length is modified, and a multiplexing header extraction process is performed by $20 \times N$ byte cycle.
 20 The number in a parenthesis is one of the examples. The state is repeated until a confirmation response of frame length modification is transmitted in the receiving end. In 502, the process goes to a mode for performing a multiplexing header extraction process by $20 \times L$ byte in the case of Yes. In 503, variable s is initialized, the variable s for protecting synchronization.
 25 variable m is initialized, m which extracts a multiplexing header interval of $20 \times M$ byte. In 505, a multiplexing header process is performed after $20 \times L$ byte. In 506, if no multiplexing header exists, m is incremented in 507. In 506, when a multiplexing header is extracted, the field placed in between the last extracted multiplexing header and the extracted multiplexing

header is extracted as a header and an information field in 508, and a header and an information field are transmitted to a higher layer. In 509, it is determined whether the interval between multiplexing headers is $20 \times M$ byte or not. And, when the interval between multiplexing headers is not $20 \times M$ byte, variable s is initialized, the variable s for protecting synchronization, and variable m is initialized, the variable m for extracting a multiplexing header interval in 504, a multiplexing header extraction process of $20 \times L$ byte continues in 505. In 509, when a multiplexing header interval is $20 \times M$ byte, a multiplexing header interval is checked for protecting synchronization in 510. S is the predetermined extraction number of a successive multiplexing header for protecting synchronization, and when s is less than S , s is incremented in 512, variable m is initialized, the variable m for extracting a multiplexing header interval in 504, and a synchronization flag extraction process of $20 \times L$ byte continues in 505. If $s=S$ in 510, it is determined that synchronization of a frame length is established, the frame length after being modified, and the process goes to reception 513 of $20 \times M$ byte cycle.

According to the embodiment 2, a frame length is modified promptly by small throughput in the case of modifying a frame length of a semi-fixed multiplexing frame even without a synchronization flag.

If the time for responding delay time is understood(e.g. MODS ARQ), the delay time from which a confirmation response is transmitted to which a frame length after the modification is transmitted, the throughput is decreased when a synchronization flag process is modified to the process per 20 byte, or per $20 \times L$ byte after the response delay time from the time of transmitting a response delay time. Further, the case in $F=20$ is described in this embodiment, however, the invention has the same effect if F is other values, and if N and M are other values not described in the above case.

Further, a mode exists in H.223Annex, the mode for interleaving a

multiplexing header in a multiplexing frame for the purpose of decreasing an error. However, it is impossible to return from an interleave mode if a frame length is not known, therefore, it is necessary to set to a mode which does not interleave a multiplexing header in advance when the
5 embodiments are carried out.

[Embodiment 3]

Fig.6 is a timing figure for indicating principle of operation of a multiplexing header extraction method according to the embodiment 3.

Fig.6 shows the principal of operation of a multiplexing header
10 extraction method when sound data, video data, other data or any combinations of sound data, video data or other data are multiplexed by making a frame length of a multiplexing frame semi-fixed, using a clock synchronization, and a multiplexing frame without having a synchronization flag in digital network after a frame synchronization is
15 established by using a synchronization frame of $20 \times N$ byte (N : natural number) having a synchronization flag. Fig.7 is a flowchart indicating the operation a multiplexing frame method according to the embodiment 3 of the present invention.

Fig.8 is a flowchart indicating the operation a multiplexing frame
20 method according to the embodiment 3 of the present invention.

Hereafter, the operation will be described by using a timing figure in fig.6, and a flowchart by using fig.7 and fig.8.

Fig.6 (a) shows a multiplexing header extraction method in the case that the network is in a normal condition, and a multiplexing header is
25 extracted at the first time if a header exists after $20 \times N \times 8$ bytes on the basis of a multiplexing header extracted last time. With regard to the determination whether it is a multiplexing header or not, for example, in the case of H.223/AnnexA, an error detection of a multiplexing information unit is performed by an error detection code unit after a multiplexing

information unit and an error of a detection code unit are corrected by using a correction code of an error correction code unit in fig.10, and it is determined as a multiplexing header if no error is detected, and the remaining numbers of byte, after which numbers of byte of a multiplexing header unit is subtracted from the next $20 \times N$ byte, is cut as information field, and multiplexing synchronization is performed on the basis of a multiplexing information unit. Fig.6 (b) shows that byte dropouts are occurred after extracting the last multiplexing header. Byte dropouts are occurred by an error of clock precision of an exchange. In this case, a header is not positioned at the header reference position after $20 \times N \times 8$ bytes from a multiplexing header extracted last time, but is positioned ahead of the number of byte dropouts. On the contrary, when byte insertions are occurred, a multiplexing header is positioned behind the number of bytes from the header reference position. The existing establishment of a header is the highest at the header reference position, and gets lower as a header gets farther from the header reference position. It is also necessary to decrease the number of times of a extraction process of a multiplexing header as the extraction process is quite complicated. Therefore, the extraction process is different from a synchronization flag extraction process in which the pattern matching is performed by shifting bytes in the receiving order of bytes, but the extraction process is started from the header reference position, and performed in the order of ± 1 byte, ± 2 byte,..... $\pm P$ byte. In this case, P is a natural number, and pre-determined, or determined timely in accordance with error occurrence of byte shifts in the network. Also, $P=0$ means that a header extraction process is performed only the header reference position. Consequently, the header extraction process is performed in descending order of the existing establishment of a correct header, the multiplexing header extraction process is cancelled after a multiplexing header with no error is extracted,

and as a result, the only one multiplexing header extraction process is necessary in the case that byte shifts do not occur. Therefore, throughput is considerably decreased. Further, since a multiplexing header reaches at intervals of $20 \times N$ byte for a while after byte shifts occur, therefore, a header is always shifted from the header reference position, before which byte shifts occur. This means it is necessary to perform a header extraction process a plurality of times although byte shifts do not occur, and when byte shifts occur to the same direction a plurality of times, a header is shifted from $\pm P$ window. Consequently, the above problem is solved by that the header reference position is determined as the position at which a multiplexing flag is extracted when a phase is performed S times, the phase between the position at which a multiplexing flag is extracted and the header reference position. In this case, S is a natural number, and pre-determined, or determined timely in accordance with error occurrence of byte shift in the network.

The above operation will be described by using a flowchart in Fig.7. In fig.7, a default value of variable s is set in 701,702. A multiplexing header extraction process is performed at a offset position by p from a header reference position in 703, 704. It is determined whether a multiplexing header is or not in 705, it is determined whether a offset value P is the last value of a offset window if a multiplexing header is not extracted, and if a offset value P is not the last one, p value is set in 707, the p value which is the next offset value. In 707, if p value one before offset value p is 0 or fewer than 0, the upper formula is used, and if p value one before offset value p is the positive integer, the lower formula is used. If $p=P$ in 706, it indicates that a header is not extracted within offset value set by using the reference number. It occurs that when it is shifted from a range of a search window by byte shifts, or when a transmission error to a multiplexing header is occurred, the error beyond the error correction capability.

From 709 to 714, it is processed that the header reference position is determined as the position at which a multiplexing flag is extracted when a phase is performed S times, the phase between the position at which a multiplexing flag is extracted and the header reference position. In 709, it is determined whether the extracted offset value is the same as that of the last time. If same, counter s is incremented in 710, it is determined whether s has reached the specified number S in 711. counter s is cleared back to zero if offset value is not subsequent in 712. Offset value p is cleared back to zero after setting offset value to p2 in 714. In 715, the next reference position is determined, and p value in 715 is added only when the reference position is determined to be shifted. In 708, the remaining numbers of byte, after which numbers of byte of a multiplexing header unit is subtracted from the next 20 x N byte, is cut as information field on the basis of a multiplexing header in which an error is not detected, and a multiplexing separation is performed in compliance with a multiplexing information unit as a multiplexing process.

Next, when a header is shifted from a range of a search window by byte shifts, a multiplexing header cannot be extracted by the above-described process. Therefore, when a multiplexing header is not extracted E1 time (E1 is a natural number) in succession from a range of $\pm P$ bytes of the reference point, it is assumed that synchronization shifts occur and the process is performed for establishing frame synchronization by using a synchronization frame of 20 x N bytes (N is a natural number) having a synchronization flag. Also, when a transmission error to a multiplexing header is occurred, the error beyond the error correction capability, a multiplexing header is not extracted at the reference position. However, a multiplexing header is still not extracted by searching on or about the reference position by using a large amount of throughput. Therefore, only a multiplexing header is performed at the reference position as $P=0$ in a

normal condition, and as a result, when a multiplexing header is not extracted E2 times (E is a natural number) in succession, it is set as $P \geq 1$. As a result, when the extracted position of a multiplexing header is shifted from the reference position for p bytes (P is an integer) S times (S is a natural number), the extraction process of a multiplexing header is performed by small amount of throughput without being effected by a transmission error by that the reference position for the next multiplexing frame is shifted for p bytes, and the multiplexing header extraction process is moved to the normal condition as $P=0$.

The above operation will be described with reference to a flowchart in fig.8. The operation in fig.8 is the same as that of fig.7 except for after 816. In 801, the size of a search window is $P=0$, and the process goes to 816 when a multiplexing header is not extracted at the reference position. It is determined whether the size of a search window is 0 or P1 in 816. P1 is a natural number and pre-determined, or determined timely. e2 is a counter for counting how many times a multiplexing header is not extracted in succession, and e2 is incremented in 818. In 820, it is determined whether the number of times, which a multiplexing header is not extracted in succession, has reached E2 set in advance or timely. When e2 has reached E2, $P=p1$ is set in 823, and a search window is enlarged, the search window for a multiplexing header extraction process. e2 is a counter for counting how many times a multiplexing header is not extracted in succession after a search window is enlarged, and e2 is incremented in 817. In 819, it is determined whether the number of times, which a multiplexing header is not extracted in succession, has reached E1 set in advance or timely after a search window is enlarged. As a result of 819, when the number of times, which a multiplexing header is not extracted in succession, has reached E1 set in advance or timely after a search window is enlarged, the process is ended in 812, and the process is performed by using a synchronization

frame of $20 \times N$ byte (N is a natural number) having a synchronization flag, the process for establishing a frame synchronization.

According to the embodiment, a multiplexing header is extracted with stability by small amount of throughput in the case that a transmission error occurs, therefore, a transmission with high efficiency is performed, the transmission without a flag and the frequency is decreased, the frequency which occurs resynchronization.

[Embodiment 4]

Fig.17 is a timing figure when a multiplexing frame having a synchronization flag is modified to a multiplexing frame without a synchronization flag according to the embodiment 4 of the present invention. A radio slot of a PHS is 20 bytes, and $F=20$ is suitable. Fig.17(a) is a timing figure of when a confirmation response transmitted by the reception side is transmitted to the transmission side at one try, the confirmation response for responding a modification request of a synchronization flag. Fig.17(b) is a timing figure of when a confirmation response transmitted by the reception side is not correct, the confirmation response for requesting to modify a synchronization flag, and the second confirmation response is transmitted to the transmission side, the second confirmation response for requesting to modify a synchronization flag and retransmitted to the transmission side as a result of time-out of confirming a confirmation response. Hereafter, the operation will be described by using the above-described timing figures and a flowchart in fig.18.

In fig.17, the field written as synchronization is a synchronization flag of a multiplexing frame it is assumed that a synchronization flag is 2 bytes. The field written as MC is a multiplexing code, and MC is protected by using an error detection code and an error correction code for a transmission error and is also 2 bytes. A code may be selected such as the code length for determining synchronization flag length or error detection

capability, or the code for determining an error correction capability. In fig.1(a), a frame having a synchronization flag is received in a first frame, and a synchronization flag is extracted and a multiplexing header after a synchronization flag is extracted, and information field after a multiplexing header is cut. With regard to extracting a synchronization flag, pattern matching may be used by shifting a synchronization flag on or about bytes, or by cutting a synchronization flag (e.g. 2 bytes) at the front. A request is stored in the information field of the first frame in fig.1, the request for modifying a synchronization flag from the second byte. The reception side for receiving a request determines whether it is possibly performed or not, and returns a confirmation response at notice. In fig.1, a modified multiplexing frame is transmitted to the reception side at the third frame, which is 2 frames after returning a confirmation response, however, the timing is not known in the reception side, the timing for transmitting a multiplexing frame. Therefore, after the reception side returns a confirmation response, a frame is received in the reception side, the frame having a synchronization flag, and the frame in which a multiplexing frame is stored at the front and which does not have a synchronization flag. A multiplexing header and information field are cut by receiving a frame having a synchronization flag as the second frame is a multiplexing frame having a synchronization flag. A multiplexing frame and information field are not extracted in the reception side of a frame having a synchronization flag as a multiplexing frame is modified to a frame without a synchronization flag in the third frame. On the other hand, a multiplexing header is extracted in the reception side of a frame starting with a multiplexing frame. In this case, when a frame is extracted S=2times in succession in the reception side, the frame starting with a multiplexing header, only a frame is received in the fifth frame, the frame starting with a multiplexing header since only a frame is received, the frame starting with

a multiplexing header after $S=2$ times in succession. In fig.1(b), the operation until the second frame is the same as that of fig.1, however, the third frame is a frame having a synchronization flag as a confirmation response transmitted by the reception side first time is not transmitted to the transmission side by a transmission error. In this case, a multiplexing header and information field are cut by receiving a frame having a synchronization flag in the reception side. Afterwards, a modification request is retransmitted in the fourth frame in the transmission side as a timer works, the timer for waiting for a confirmation response, and the timer being set when a modification request is transmitted first in the transmission side. In this case, a modification request retransmitted is received in the reception side since a multiplexing header and information field are cut by receiving a frame having a synchronization flag. In fig.(b), a second confirmation response is transmitted to the transmission side from the reception side, the second confirmation response for the retransmission of a modification request, therefore, the operation in fig.(b) after the fifth frame is the same as that of fig.(a) after the second frame.

A flowchart describing the above operation is shown in fig.18. In fig.18, 1801 indicates state prior to modifying a synchronization flag, and synchronization flag extraction process is performed by $20 \times N$ byte cycle in 1801. The number in a parenthesis is one of the examples. The state in 1801 is repeated until the reception side returns a confirmation response of synchronization flag modification. When Yes in 1802, the process goes to the one for extracting the first 2 bytes by $20 \times N$ byte. In 1803, variable s is initialized, the variable s for synchronization protection. In 1804, the process is performed, the process for extracting the first 2 bytes after $20 \times N$ bytes. In 1805, when 2 the first 2 bytes are not a synchronization flag, the process goes to 1807. Also, in 1805, when a synchronization flag is not extracted, the field after 2 bytes is extracted as a header and information

field, and a header and information field are transmitted to a higher layer. As a result of analyzing a synchronization flag in the higher layer, when the received header is a modification request of a synchronization flag, the higher layer returns a confirmation response. In 1807, a multiplexing header in the first 2 bytes is extracted. With regard to extracting a multiplexing header, after an error correction is performed to a multiplexing header by the attached error detection code, if no error is detected as a result of an error detection by an error detection code, the first 2 bytes are assumed as a multiplexing header in 1808. Consequently, if the first 2 bytes are not determined as a multiplexing header in 1808, s is reset in 1809, and the process goes to receiving the next frame. On the other hand, if the first 2 bytes are a multiplexing header, it is determined whether the consecutive numbers of extracting a multiplexing header has reached the predetermined value S in 1810. If $s=S$, a multiplexing header is received without a synchronization flag after 1812. If $s < S$, s is incremented in 1811, and the process goes to receiving the next frame.

According to the embodiment, when a multiplexing frame is modified, from the multiplexing frame having a synchronization flag to multiplexing frame without a synchronization flag, a synchronization flag is modified without fault and swiftly.

Further, as MODS ARQ, when the delay time is known in advance, the delay time from which a confirmation response is transmitted to that the modified frame length is returned as a result of a confirmation response, if the process goes to determining whether the first 2 bytes are a multiplexing header at the time after the response delay time is over from the time that the response delay time is returned, consequently, throughput is even more decreased. Further, $F=20$ is described in the embodiment, however, although F value is the other values, the present invention has the same effect.

In the above-case, a synchronization flag and a multiplexing header are 2 bytes. However, there is nothing wrong that a synchronization flag and a multiplexing header are not always 2 bytes, or each of a synchronization flag and a multiplexing header has a different number of byte. In this embodiment, a synchronization flag and a multiplexing header are extracted separately in 1804, a synchronization flag is determined in 1805, and a multiplexing header is determined in 1807 and 1808. Further, in this embodiment, the case is described that a multiplexing frame having a synchronization flag is modified to a multiplexing frame without a synchronization flag, but a multiplexing frame without a synchronization flag is modified to a multiplexing frame having a synchronization flag. In this case, a frame without a synchronization flag is received in 1801, and a frame having a synchronization flag is received in 1812.

Further, with regard to the case that a synchronization flag is modified and multiplexing frame length is modified to $F \times M$ byte at the same time, it is performed by that an extraction process of a multiplexing header is performed per F byte or per $F \times L$ byte (L is the greatest common divisor of N and M).

Further, for the purpose of processing easily, the reception process of a frame without a synchronization flag is performed from the middle of the reception process of a frame having a synchronization flag. However, it can be determined comprehensively after the above processes are completed. For example, in 1805, although a synchronization flag is extracted in the first field of a multiplexing frame having a synchronization flag, or although a synchronization flag is not extracted, a multiplexing header extraction is performed in the case that an error transmission is occurred in a synchronization flag when the process goes to 1807 in the case that a multiplexing header is not extracted after a header and information field are extracted in 1806, and consequently, a multiplexing header is more

strengthened to a transmission error. Further, even when a multiplexing header is extracted in 1806, the next information field is transmitted to a higher layer, and a multiplexing header without the extraction process of a synchronization flag is performed in 1807. Then, if a multiplexing header is
5 extracted in 1807, this means a multiplexing header in 1806 or 1807 is mistakenly detected, but it is impossible to specify which a multiplexing header is mistakenly detected. Therefore, variable *s* is incremented, and the information field, transmitted to a higher layer, may be detected by an error detection function in the higher layer.

10 Further, when error correction capability of a multiplexing header, or error detection capability of a multiplexing header is modified, or both capabilities of a multiplexing header are modified at the same time of modifying a synchronization flag, a confirmation response is transmitted to each above-modification command from the transmission side. Then, when
15 a frame having a synchronization flag and a frame without a synchronization flag are received in the reception side after transmitting a confirmation response, or predetermined time, with regard to receiving a frame having a synchronization flag, a header extraction in 1806 is performed by using error correction capability and error detection
20 capability both of which are not modified, and with regard to receiving a frame without a synchronization flag, a header extraction in 1806 is performed by using error correction capability and error detection capability both of which are modified. Consequently, the above capabilities are modified along with the modification of a synchronization flag.

25 Further, when error correction capability of a multiplexing header, or error detection capability of a multiplexing header is modified, or both capabilities of a multiplexing header are modified without modifying a synchronization flag, it is not determined in 1805. Then, a multiplexing header at the specified position is extracted by using a parameter before

JP1998-178406

and after modification, and when a multiplexing header of a parameter after the modification is extracted S times in succession, the above capabilities consequently are modified.

Further, without modifying an error correction capability, the above
5 capabilities are modified.

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平10-178406

(43)公開日 平成10年(1998)6月30日

(51)Int.Cl. ⁵	識別記号	F I
H 0 4 J 3/00		H 0 4 J 3/00 B
3/06		3/06 A
H 0 4 L 7/08		H 0 4 L 7/08 A

審査請求 未請求 請求項の数33 O L (全 19 頁)

(21)出願番号 特願平8-337887

(22)出願日 平成8年(1996)12月18日

(71)出願人 000005821

松下電器産業株式会社

大阪府門真市大字門真1006番地

(72)発明者 黒部 彰夫

大阪府門真市大字門真1006番地 松下電器
産業株式会社内

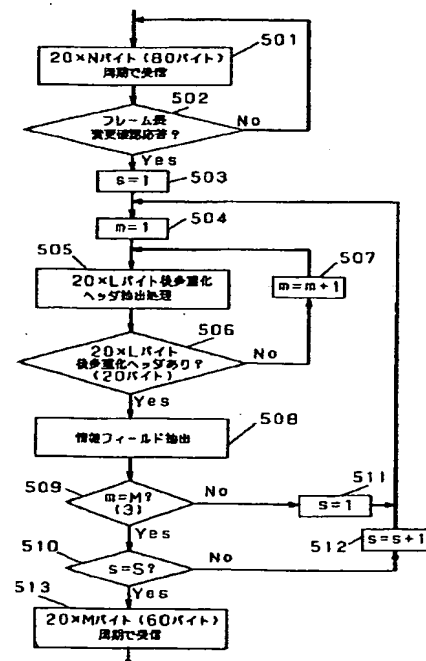
(74)代理人 弁理士 滝本 智之 (外1名)

(54)【発明の名称】 多重化フレーム長変更方法ならびに同期フラグ変更方法および多重化ヘッダ抽出方法

(57)【要約】

【課題】 半固定長の多重化フレームで音声、ビデオを多重伝送する場合の多重化フレーム長を多重化フレームの先頭に同期フラグがある場合もない場合も容易にフレーム長の変更を可能とする。

【解決手段】 同期フレームを用いてフレーム同期を確立した後、多重化フレームのフレーム長を半固定にし、デジタル網のクロック同期を利用することにより同期フラグを有しない多重化フレームであって、送信側が半固定のフレーム長を $20 \times N$ バイト (N は自然数) から $20 \times M$ バイト (M は N 以外の自然数) に変更する場合において、受信側で多重化ヘッダ抽出処理を $20 \times L$ (L は M と N の最大公約数) バイトおきに実行する。



【特許請求の範囲】

【請求項1】同期フラグを有する多重化フレームのフレーム長を半固定にして音声データまたはビデオデータまたはその他のデータまたはそれらの任意の組み合わせを多重化して伝送する方法であって、送信側が前記半固定のフレーム長を $F \times N$ バイト（ F 、 N は自然数）から $F \times M$ バイト（ M は N 以外の自然数）に変更する場合において、受信側で同期フラグ抽出処理を F バイトおきに実行することを特徴とした多重化フレーム長変更方法。

【請求項2】同期フラグを有する多重化フレームのフレーム長を半固定にして音声データまたはビデオデータまたはその他のデータまたはそれらの任意の組み合わせを多重化して伝送する方法であって、送信側が前記半固定のフレーム長を $F \times N$ バイト（ F 、 N は自然数）から $F \times M$ バイト（ M は N 以外の自然数）に変更する場合において、受信側で同期フラグ抽出処理を $F \times L$ バイト（ L は M と N の最大公約数）おきに実行することを特徴とした多重化フレーム長変更方法。

【請求項3】送信側から多重化フレーム長変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側は前記確認応答受信後に多重化フレーム長を変更する場合に、受信側は確認応答返送後、同期フラグ抽出処理を F バイトおきまたは $F \times L$ バイトおきに実行することを特徴とした請求項1または請求項2記載の多重化フレーム長変更方法。

【請求項4】送信側から多重化フレーム長変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側はあらかじめ前記応答に要する応答遅延時間を測定しており、送信側は前記応答遅延時間の測定結果を受信側に通知しており、送信側は前記確認応答受信後に多重化フレーム長を変更する場合に、受信側は確認応答返送時から前記応答遅延時間経過後、同期フラグ抽出処理を F バイトおきまたは $F \times L$ バイトおきに実行することを特徴とした請求項1または請求項2記載の多重化フレーム長変更方法。

【請求項5】 F バイトおき、または $F \times L$ バイトおきに実行した同期フラグ抽出処理の結果、 $F \times M$ バイトおきに連続してあらかじめ決められた回数以上の同期フラグが抽出された場合に、以降の受信側での同期フラグの抽出処理を $F \times M$ バイトおきにすることとしたことを特徴とした請求項1から請求項4のいずれか1項に記載の多重化フレーム長変更方法。

【請求項6】同期フラグを有する $F \times N$ バイトの同期フレームを用いてフレーム同期を確立した後、多重化フレームのフレーム長を半固定にし、デジタル網の同期を利用することにより同期フラグを有しない多重化フレームにより音声データまたはビデオデータまたはその他のデータまたはそれらの任意の組み合わせを多重化して伝送する方法であって、送信側が前記半固定のフレーム長を $F \times N$ バイト（ F 、 N は自然数）から $F \times M$ バイト

（ M は N 以外の自然数）に変更する場合において、受信側で多重化ヘッダ抽出処理を F バイトおきに実行することを特徴とした多重化フレーム長変更方法。

【請求項7】同期フラグを有する $F \times N$ バイトの同期フレームを用いてフレーム同期を確立した後、多重化フレームのフレーム長を半固定にし、デジタル網の同期を利用することにより同期フラグを有しない多重化フレームにより音声データまたはビデオデータまたはその他のデータまたはそれらの任意の組み合わせを多重化して伝送する方法であって、送信側が前記半固定のフレーム長を $F \times N$ バイト（ F 、 N は自然数）から $F \times M$ バイト

（ M は N 以外の自然数）に変更する場合において、受信側で多重化ヘッダ抽出処理を $F \times L$ （ L は M と N の最大公約数）バイトおきに実行することを特徴とした多重化フレーム長変更方法。

【請求項8】送信側から多重化フレーム長変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側は前記確認応答受信後に多重化フレーム長を変更する場合に、受信側は確認応答返送後、多重化ヘッダ抽出処理を F バイトおきまたは $F \times L$ バイトおきに実行することを特徴とした請求項6または請求項7記載の多重化フレーム長変更方法。

【請求項9】送信側から多重化フレーム長変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側はあらかじめ前記応答に要する応答遅延時間を測定しており、送信側は前記応答遅延時間の測定結果を受信側に通知しており、送信側は前記確認応答受信後に多重化フレーム長を変更する場合に、受信側は確認応答返送時から前記応答遅延時間経過後、多重化ヘッダ抽出処理を F バイトおき、または $F \times L$ バイトおきに実行することを特徴とした請求項6または請求項7記載の多重化フレーム長変更方法。

【請求項10】前記多重化ヘッダ抽出処理により抽出された多重化ヘッダではさまれた領域を情報フィールドとみなして上位層に渡すことを特徴とした請求項6から請求項9のいずれか1項に記載の多重化フレーム長変更方法。

【請求項11】前記多重化フレーム長変更コマンドを送信する以前に多重化ヘッダをインタリーブしないモードに変更しておくことを特徴とした請求項6から請求項10のいずれか1項に記載の多重化フレーム長変更方法。

【請求項12】同期フラグの抽出処理として受信データを1ビットずつシフトしながら予め決められたビット数の同期パターンと受信データを比較してパターンマッチングを行うことを特徴とした請求項1から請求項5のいずれか1項に記載の多重化フレーム長変更方法。

【請求項13】多重化ヘッダの抽出処理として多重化ヘッダに添付された誤り訂正符号を用いて多重化情報部と誤り検出符号の誤り訂正を行った後、誤り検出符号を用いて多重化情報部の誤り検出をし、誤りが検出されない

10

20

30

40

50

場合に、多重化ヘッダとして抽出することを特徴とした請求項6から請求項11のいずれか1項に記載の多重化フレーム長変更方法。

【請求項14】 $F=20$ 、 $N=4$ 、 $M=2$ 、 $L=2$ であることを特徴とした請求項2から請求項4、または請求項7から請求項12のいずれか1項に記載の多重化フレーム長変更方法。

【請求項15】 $F=20$ 、 $N=4$ 、 $M=3$ 、 $L=1$ であることを特徴とした請求項2から請求項4、または請求項7から請求項12のいずれか1項に記載の多重化フレーム長変更方法。

【請求項16】 $F=20$ 、 $N=4$ 、 $M=6$ 、 $L=2$ であることを特徴とした請求項2から請求項4、または請求項7から請求項12のいずれか1項に記載の多重化フレーム長変更方法。

【請求項17】 $F=20$ 、 $N=4$ 、 $M=2$ であることを特徴とした請求項1または請求項6に記載の多重化フレーム長変更方法。

【請求項18】 $F=20$ 、 $N=4$ 、 $M=3$ であることを特徴とした請求項1または請求項6に記載の多重化フレーム長変更方法。

【請求項19】 $F=20$ 、 $N=4$ 、 $M=6$ であることを特徴とした請求項1または請求項6に記載の多重化フレーム長変更方法。

【請求項20】同期フラグを有する $F \times N$ バイト (F 、 N は自然数)の同期フレームを用いてフレーム同期を確立した後、多重化フレームのフレーム長を半固定にし、デジタル網の同期を利用することにより同期フラグを有しない多重化フレームにより音声データまたはビデオデータまたはその他のデータまたはそれらの任意の組み合わせを多重化して伝送する方法の多重化ヘッダ抽出方法であって、ひとつ前の多重化フレームの多重化ヘッダの基準位置から $F \times N$ バイト後を次の多重化フレームの基準位置とし、基準位置の前後 P ビット (P は整数)の範囲で多重化ヘッダの抽出処理を行うことを特徴とした多重化ヘッダ抽出方法。

【請求項21】まず始めに基準位置の多重化ヘッダ抽出処理を行い、次に基準位置の前後 J ビットシフトした範囲で多重化ヘッダの抽出処理を行い、以下順次基準位置の前後 P ビット (P は整数)シフトした範囲まで J ビットきざみで多重化ヘッダの抽出処理を行うことを特徴とした請求項20記載の多重化ヘッダ抽出方法。

【請求項22】多重化ヘッダの抽出処理の結果、ヘッダ情報に誤りのない多重化ヘッダが抽出され次第、その基準位置での以下の多重化ヘッダ抽出処理は中断することを特徴とした請求項21記載の多重化ヘッダ抽出方法。

【請求項23】抽出されたヘッダ以下の $F \times N$ バイトを情報フィールドとみなして多重分離を行うことを特徴とした請求項20から請求項22のいずれか1項に記載の多重化ヘッダ抽出方法。

【請求項24】抽出された多重化ヘッダの位置が基準位置から p ビット (p は整数)シフトする状態が S 回 (S は自然数)連続した場合、次の多重化フレームの基準位置を p ビットシフトすることを特徴とした請求項20から請求項23のいずれか1項に記載の多重化ヘッダ抽出方法。

【請求項25】基準位置の前後 P ビットの範囲から多重化ヘッダが抽出されない状態が $E1$ 回 ($E1$ は自然数)連続した場合に同期外れとみなして同期フラグを有する $F \times N$ バイト (F 、 N は自然数)の同期フレームを用いてフレーム同期を確立する手順を実行することを特徴とした請求項20から請求項24のいずれか1項に記載の多重化ヘッダ抽出方法。

【請求項26】正常時には $P=0$ として基準位置のみの多重化ヘッダ抽出処理を行っており、その結果、多重化ヘッダが抽出されない状態が $E2$ 回 ($E2$ は自然数)連続した場合に $P \geq 1$ に設定し、その結果、抽出された多重化ヘッダの位置が基準位置から p ビット (p は整数)シフトする状態が S 回 (S は自然数)連続した場合、次の多重化フレームの基準位置を p ビットシフトし、 $P=0$ とした正常状態に移行することを特徴とした請求項20から請求項25のいずれか1項に記載の多重化ヘッダ抽出方法。

【請求項27】同期フラグを有する $F \times N$ バイト (F 、 N は自然数)の多重化フレームから同期フラグを有しない $F \times N$ バイトの多重化フレームに移行する場合に、同期フラグを有する多重化フレームの受信と並行して、多重化ヘッダ抽出処理を $F \times N$ バイト毎に行うことを特徴とした同期フラグ変更方法。

【請求項28】同期フラグを有する $F \times N$ バイト (F 、 N は自然数)の多重化フレームから同期フラグを有しない $F \times M$ バイト (M は N 以外の自然数)の多重化フレームに移行する場合に、同期フラグを有する多重化フレームの受信と並行して、多重化ヘッダ抽出処理を $F \times N$ バイト毎に行い、 F バイト毎または $F \times L$ バイト毎 (L は N と M の最大公約数)に多重化ヘッダ抽出処理を行うことを特徴とした同期フラグ変更方法。

【請求項29】送信側から同期フラグ変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側は前記確認応答受信後に同期フラグを変更する場合に、受信側は確認応答返送後、多重化ヘッダ抽出処理を $F \times N$ バイトおきまたは F バイトおきまたは $F \times L$ バイトおきに実行することを特徴とした請求項27または請求項28記載の同期フラグ変更方法。

【請求項30】送信側から同期フラグ変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側はあらかじめ前記応答に要する応答遅延時間を測定しており、送信側は前記確認応答遅延時間の測定結果を受信側に通知しており、送信側は前記確認応答受信後に同期フラグを変更する場合に、受信側は確認応答返送時から前記

応答遅延時間経過後、多重化ヘッダ抽出処理をF×NバイトおきまたはFバイトおきまたはF×Lバイトおきに実行することを特徴とした請求項27から請求項30のいずれか1項に記載の同期フラグ変更方法。

【請求項31】多重化フレームの先頭に多重化ヘッダがS回連続して抽出されるまでは、同期フラグを有する多重化フレームの受信を並行して行うことを特徴とした請求項27から請求項30のいずれか1項に記載の同期フラグ変更方法。

【請求項32】多重化ヘッダの誤り訂正能力または誤り検出能力またはその双方を同期フラグの変更と同時に変更する場合において、前記同期フラグを有する多重化フレームの受信は変更前の誤り訂正能力および誤り検出能力で多重化ヘッダの誤り訂正および誤り検出を行い、前記多重化フレームの先頭にて行う多重化ヘッダ抽出処理は変更後の誤り訂正能力および誤り検出能力で多重化ヘッダの抽出処理を行うことを特徴とした請求項27から請求項31のいずれか1項に記載の同期フラグ変更方法。

【請求項33】前記同期フラグおよび前記多重化ヘッダ以外のパラメータの変更は前記多重化フレームの先頭に多重化ヘッダがS回連続して抽出された時以降に行うことを特徴とした請求項31または請求項32記載の同期フラグ変更方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、高能率圧縮された音声およびビデオデータとコンピュータデータを多重して伝送するマルチメディア通信システムの多重伝送、特にその多重化フレーム長の変更および多重化ヘッダの抽出に関する。

【0002】

【従来の技術】テレビ電話やテレビ会議システムにおいては、膨大な情報量を持つビデオ信号は高能率で符号化した後、伝送先に伝送することが一般的である。このような高能率符号化の例としてはITU-Tの国際標準規格であるH.261やH.263がある。これらの圧縮原理はいずれも動き補償フレーム間予測符号化方法や可変長符号化であり、伝送誤りに対しては大変敏感であり1ビットの誤りでも画面の大きな範囲へ伝搬するとともに時間方向にも伝搬し、画質の劣化が著しいため、伝送誤りが発生しやすいアナログ電話回線では再送制御により誤り訂正を行ってから画像復号化を行うのが一般的である。一方音声データもITU-Tで勧告されたG.723のように高能率符号化により超低ビットレートに圧縮して伝送するが、ビデオデータに比べると伝送誤りが目立ちにくい性質をもつ反面、伝送遅延には敏感であり、よって再送による誤り訂正は用いず、誤りを検出した場合にはミューティングによりノイズ化することを防止している。G.723の伝送レートは5.3Kbps

または6.3Kbpsであり、30ms毎に20バイト程度の音声フレームを発生する。これを32Kbpsの伝送速度を持つPHSで伝送する場合120バイト周期で1フレームとなる。こうしたビデオデータと音声データを多重伝送する方法として同じくITU-Tの勧告としてH.223が発行されている。H.223は当初、有線での通信を対象に考えられていたが、現在、H.223をさらに無線通信を対象に拡張したものとして、H.223/AnnexAが提案されている。このH.223/AnnexAは、誤りが多く、かつ、誤りがバースト的に発生する無線伝送路の特徴を考慮して、H.223からいくつかの変更がなされている。

【0003】図9に、H.223/AnnexAの多重化層の多重化フレームフォーマットを示す。同期フラグのあと、固定長のヘッダと、映像、音声、データが多重された情報フィールドが続く。同期フラグのサイズは選択できるが、初期状態では4バイトの同期フラグ「1111100011011101010000100101100」を用いることになっている。

【0004】図10に、ヘッダのフォーマットを示す。ヘッダ情報は、情報フィールドにどの種類のデータがどのように多重されているかを示す多重化情報部と、この多重化情報部につけられるCRCである誤り検出符号部と誤り訂正符号部からなる。多重化フレームのフレーム長や多重化フレームの多重化の構造は決められておらず、通信時に送信側から構成情報として通知する。多重化構造については最大で16種類までの多重構造を定義してヘッダ情報と1対1に対応付けたテーブルを受信側へ転送しておき、受信側は多重化フレームを受信する毎にヘッダを解析し、情報フィールドのどの領域にどんなデータが格納されているかを知るしくみになっている。

【0005】一方、従来、アナログ電話回線に比べさらに伝送誤りの発生しやすい32Kbpsの伝送速度を持つPHSの無線回線においてコンピュータデータを伝送する際の伝送制御方式として、セレクトティブリビートARQの一種である、MODS ARQ(SRARQ with Modulo Operation using Data field)が提案されている。このMODS ARQについては、1995年電子情報通信学会通信総合大会B-495、1995年電子情報通信学会通信通信ソサイエティ大会B-318、1996年電子情報通信学会通信総合大会B-470において発表されている。

【0006】MODS ARQにおけるARQフレームは固定長である。図11にMODS ARQのフレーム構成を示す。フレーム種別識別領域にはフレーム種別が、ユーザデータ長表示領域には、ユーザデータ領域内の有意なデータのデータ長が、ユーザデータ領域にはユーザデータが、フレーム番号表示領域には送信フレーム番号が、要求フレーム番号表示領域には最旧未受信フレーム番号が、誤り検出符号領域には、ARQフレーム全体の誤りを検出する誤り検出符号が格納されている。フレーム同期は通信に先だ

って同期フラグを含んだ同期フレームを用いて確立し、データフレームのフレーム長を同期フレームと同じとすることで以降のフレーム同期を継続するため、データフレームに同期フラグを必要としない。

【0007】クロック同期のとれたデジタル網であるPHS網ではビット同期が保証されるため、MODS-ARQではデータ通信に先だって、同期フラグ「01010000111011110010100110010011」を含む同期フレームを用いてフレーム同期を確立した後は、同期フレームと同じフレーム長の固定長のデータフレームを用いることにより、フレーム同期を維持しながらデータ通信を行う。このデータフレームには同期フラグは不要である。80バイトのフレームに4バイトの同期フラグを常に付加していると、効率を5%低下させることになる。

【0008】そこでMODS-ARQの同期フレームで同期を確立した後に、同期フレームと同じフレーム長の固定長の多重化フレームを用いることにより、H.223/AnnexAの多重化層の多重化フレームフォーマットから同期フラグを削除して効率を上げる方法を我々は提案している（特願平8-187153号「多重伝送方法および分離方法」）。図12にこの場合の多重化フレームフォーマットを示す。

【0009】そこでは、図13のように、1フレームの音声フレームを格納する領域を含む80byteの第1の固定長多重化フレームと、音声フレームを多重しない80byteの第2の固定長多重化フレームが2:1の割合で現れるパターンで伝送する形態が考えられている。

【0010】こうすることにより80バイトの多重化フレームで120バイト中1フレームの音声データを伝送できる。さらにこの場合には、無音区間において音声フレームが発生しなかったときに、第2の固定長多重化フレームを用いることでビデオフレームをロングフレームに拡張でき伝送帯域が増やせる。このロングフレームに誤りが生じた際の再送時に有音となったときでも、第2の固定長多重化フレームの固定長スロットで再送を行うことができる。しかし、この場合、図13のように音声フレームに40バイトのジッタが発生するという課題がある。G.723のように無音検出を行い、無音時には音声フレーム発生しないようにすることで、ビデオデータ等の伝送帯域を増やす機能を生かすには、ジッタの発生は数ms以下に抑えることが必要で、ジッタを吸収しないとG.723の無音機能の真価を発揮できない。

【0011】そこで、多重化フレーム構成としては図14のように1フレームの音声フレームを格納する領域を含む60byteの第1の固定長多重化フレームと、音声フレームを多重しない80byteの第2の固定長多重化フレームが交互に現れるパターンで伝送する形態が考えられている。

【0012】

【発明が解決しようとする課題】上述した、多重方法では以下の2つの課題が生じる。

【0013】第1の課題は多重化フレームのフレーム長を変更する際に生じる。H.223/AnnexAでは先頭に同期フラグを有した多重化フレームのフレーム長、同期フラグ、多重化ヘッダの誤り訂正能力および誤り検出能力を変更するコマンドおよびシーケンスが規定されている。図15にそのシーケンスを示す。図15(a)は正常な場合のシーケンスであり、送信側が多重化フレームのフレーム長を60バイトに変更する旨のコマンドを送出する。これを受けた受信側は60バイトの変更が可能な場合、確認応答を返送する。また、60バイトの変更が不可能な場合には拒絶応答を返送する。確認応答を受けた送信側はただちに送信フレームのフレーム長を60バイトに変更する。受信側には確認応答を返送してから応答遅延時間の後に60バイトのフレームが届くことになるがそのタイミングは網の遅延時間や送信装置の処理時間などによりばらつきがある。さらに、図15(b)の場合のように確認応答に伝送誤りが発生した場合には確認応答待ちのタイムアウト発生により送信側が再び多重化フレームのフレーム長を60バイトに変更する旨のコマンドを送出する。この場合、再送される80バイトのコマンドを受信する必要があるため、受信側は確認応答を返送した後は、60バイトのフレームも80バイトのフレームも受信できる必要がある。ところが、H.223/AnnexAは同期フラグの抽出処理の省略とエラー耐性強化の目的で多重化フレームを半固定長化している。半固定長の意味は前述したフレーム長変更シーケンスを実行しないかぎりにおいてはフレーム長を固定長とすることである。同期フラグの抽出処理は固定長フレーム時間後の前後数ビットに対してのみ行えばよい。しかし、図16に示すように、フレーム長が変更される場合において、変更後のフレームの到着を予測してその近傍のみの同期フラグを抽出しようとした場合、図16(a)のように予測が当たればよいが、図16(b)のように変更後のフレームの到着が1フレームでもおけると二度と同期フレームを抽出することはできない。よって、図16(c)のように広範囲におよんで同期フラグをサーチする必要がある。同期フラグの抽出処理としては受信データを1ビットずつシフトしながら予め決められたビット数の同期パターンと比較してパターンマッチングを行う必要があり膨大な処理が必要である。さらに、図16(d)のように同期フラグのない場合においては、多重化ヘッダの抽出処理として多重化ヘッダに添付された誤り訂正符号を用いて多重化情報部と誤り検出符号の誤り訂正を行った後、誤り検出符号を用いて多重化情報部の誤り検出をし、誤りが検出されない場合に、多重化ヘッダとして抽出する必要があり、これを1ビットシフトごとまたは1バイトシフトごとに行うのはさらに膨大な処理となる。だからといって、MODS-ARQの同期フレームで同

期を確立した後に、同期フレームと同じフレーム長の固定長の多重化フレームを用いることにより、H.223/AnnexAの多重化層の多重化フレームフォーマットから同期フラグを削除して効率を上げる場合に、フレーム長を変更する時だけ同期フラグをつけるようにすると、受信装置はH.223/AnnexAで規定された同期フラグとMODS-ARQの同期フレームとして規定されている同期フラグの双方を抽出する必要があり、携帯可能な装置を実現するうえでの大きな課題となる。

【0014】また、同期フラグ、多重化ヘッダの誤り訂正能力や誤り検出能力を変更する場合においても同様に、いつ変更されるかわからないという課題が生ずる。

【0015】以上の課題に対して、送信側から何フレーム後にパラメータが変更されるかを前もって通知する方法が考えられるが、この場合伝送エラーによりその情報が失われることを考えると常時、前もって複数回の通知を行う必要があり、変更に時間を要する。また、それでも想定している回数以上の伝送エラーが生じた場合には変更のタイミングがわからなくなるといった、不確実性の課題もある。

【0016】第2の課題は同期フレームで同期を確立した後に、同期フレームと同じフレーム長の固定長の多重化フレームを用いることにより、H.223/AnnexAの多重化層の多重化フレームフォーマットから同期フラグを削除して効率を上げる場合に、網の交換機相互のクロック精度の誤差により発生するビットずれにより起こる。ビットずれが発生するとフレームの位置がずれるため、固定長であることを利用してフレームの位置を検出しているとフレームの境界がわからなくなってしまう。H.223/AnnexAの同期フラグだけを削除してその他のフレーム構成は同じだとすると、多重化ヘッダの誤り検出でエラーとなり多重分離ができない状態が続くことになる。従来このような場合には予め決められた回数のエラーを検出した時点で同期はずれとみなして、再度同期フレームによる再同期の手順に移行するが、数百msから数秒の間データの伝送が中断される。これは、即時性のないデータの伝送の場合問題はないが、即時性のある音声やビデオデータの場合、大きな課題となる。

【0017】本発明は、上記のような問題を解決するためのもので、同期フラグがある場合もない場合においても少ない処理量で多重化フレーム長の変更および同期フラグの変更を確実にかつ速やかに行うことおよび同期フラグがない場合にも網のビットずれ発生に対して即座に同期を回復することを目的とする。

【0018】

【課題を解決するための手段】この課題を解決するために、本発明の多重化フレーム長変更方法および多重化ヘッダ抽出方法では、請求項1記載の発明においては、同期フラグを有する多重化フレームのフレーム長を半固定にして音声データまたはビデオデータまたはその他のデ

ータまたはそれらの任意の組み合わせを多重化して伝送する方法であって、送信側が半固定のフレーム長を $F \times N$ バイト(F , N は自然数)から $F \times M$ バイト(M は N 以外の自然数)に変更する場合において、受信側で同期フラグ抽出処理を F バイトおきに実行することとしている。

【0019】請求項2記載の発明においては、同期フラグを有する多重化フレームのフレーム長を半固定にして音声データまたはビデオデータまたはその他のデータまたはそれらの任意の組み合わせを多重化して伝送する方法であって、送信側が半固定のフレーム長を $F \times N$ バイト(F , N は自然数)から $F \times M$ バイト(M は N 以外の自然数)に変更する場合において、受信側で同期フラグ抽出処理を $F \times L$ バイト(L は M と N の最大公約数)おきに実行することとしている。

【0020】請求項3記載の発明においては、請求項1または請求項2において送信側から多重化フレーム長変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側は確認応答受信後に多重化フレーム長を変更する場合に、受信側は確認応答返送後、同期フラグ抽出処理を F バイトおきまたは $F \times L$ バイトおきに実行することとしている。

【0021】請求項4記載の発明においては、請求項1または請求項2において送信側から多重化フレーム長変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側はあらかじめ応答に要する応答遅延時間を測定しており、送信側は応答遅延時間の測定結果を受信側に通知しており、送信側は確認応答受信後に多重化フレーム長を変更する場合に、受信側は確認応答返送時から応答遅延時間経過後、同期フラグ抽出処理を F バイトおきまたは $F \times L$ バイトおきに実行することとしている。

【0022】請求項5記載の発明においては、請求項1から請求項4のいずれかにおいて F バイトおき、または $F \times L$ バイトおきに実行した同期フラグ抽出処理の結果、 $F \times M$ バイトおきに連続してあらかじめ決められた回数以上の同期フラグが抽出された場合に、以降の受信側での同期フラグの抽出処理を $F \times M$ バイトおきにすることとしている。

【0023】請求項6記載の発明においては、同期フラグを有する $F \times N$ バイトの同期フレームを用いてフレーム同期を確立した後、多重化フレームのフレーム長を半固定にし、デジタル網の同期を利用することにより同期フラグを有しない多重化フレームにより音声データまたはビデオデータまたはその他のデータまたはそれらの任意の組み合わせを多重化して伝送する方法であって、送信側が半固定のフレーム長を $F \times N$ バイト(F , N は自然数)から $F \times M$ バイト(M は N 以外の自然数)に変更する場合において、受信側で多重化ヘッダ抽出処理を F バイトおきに実行することとしている。

【0024】請求項7記載の発明においては、同期フラグを有する $F \times N$ バイトの同期フレームを用いてフレーム同期を確立した後、多重化フレームのフレーム長を半固定にし、デジタル網の同期を利用することにより同期フラグを有しない多重化フレームにより音声データまたはビデオデータまたはその他のデータまたはそれらの任意の組み合わせを多重化して伝送する方法であって、送信側が半固定のフレーム長を $F \times N$ バイト（ F 、 N は自然数）から $F \times M$ バイト（ M は N 以外の自然数）に変更する場合において、受信側で多重化ヘッダ抽出処理を $F \times L$ （ L は M と N の最大公約数）バイトおきに実行することとしている。

【0025】請求項8記載の発明においては、請求項6または請求項7において送信側から多重化フレーム長変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側は確認応答受信後に多重化フレーム長を変更する場合に、受信側は確認応答返送後、多重化ヘッダ抽出処理を F バイトおきまたは $F \times L$ バイトおきに実行することとしている。

【0026】請求項9記載の発明においては、請求項6または請求項7において送信側から多重化フレーム長変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側はあらかじめ応答に要する応答遅延時間を測定しており、送信側は応答遅延時間の測定結果を受信側に通知しており、送信側は確認応答受信後に多重化フレーム長を変更する場合に、受信側は確認応答返送時から応答遅延時間経過後、多重化ヘッダ抽出処理を F バイトおきまたは $F \times L$ バイトおきに実行することとしている。

【0027】請求項10記載の発明においては、請求項6から請求項9のいずれかにおいて多重化ヘッダ抽出処理により抽出された多重化ヘッダではさまれた領域を情報フィールドとみなして上位層に渡すこととしている。

【0028】請求項11記載の発明においては、請求項6から請求項10のいずれかにおいて多重化フレーム長変更コマンドを送信する以前に多重化ヘッダをインタリーブしないモードに変更しておくこととしている。

【0029】請求項12記載の発明においては、請求項1から請求項5のいずれかにおいて同期フラグの抽出処理として受信データを1ビットずつシフトしながら予め決められたビット数の同期パターンと受信データを比較してパターンマッチングを行うこととしている。

【0030】請求項13記載の発明においては、請求項6から請求項10のいずれかにおいて多重化ヘッダの抽出処理として多重化ヘッダに添付された誤り訂正符号を用いて多重化情報部と誤り検出符号の誤り訂正を行った後、誤り検出符号を用いて多重化情報部の誤り検出をし、誤りが検出されない場合に、多重化ヘッダとして抽出することとしている。

【0031】請求項14記載の発明においては、請求項

2から請求項4、または請求項7から請求項12のいずれかにおいて $N=4$ 、 $M=2$ 、 $L=2$ であることとしている。

【0032】請求項15記載の発明においては、請求項2から請求項4、または請求項7から請求項12のいずれかにおいて $N=4$ 、 $M=3$ 、 $L=1$ であることとしている。

【0033】請求項16記載の発明においては、請求項2から請求項4、または請求項7から請求項12のいずれかにおいて $N=4$ 、 $M=6$ 、 $L=2$ であることとしている。

【0034】請求項17記載の発明においては、請求項1または請求項6において $F=20$ 、 $N=4$ 、 $M=2$ であることとしている。

【0035】請求項18記載の発明においては、請求項1または請求項6において $F=20$ 、 $N=4$ 、 $M=3$ であることとしている。

【0036】請求項19記載の発明においては、請求項1または請求項6において $F=20$ 、 $N=4$ 、 $M=6$ であることとしている。

【0037】請求項20記載の発明においては、多重化ヘッダ抽出方法として、ひとつ前の多重化フレームの多重化ヘッダの基準位置から $F \times N$ バイト後を次の多重化フレームの基準位置とし、基準位置の前後 P ビット（ P は整数）の範囲で多重化ヘッダの抽出処理を行うこととしている。

【0038】請求項21記載の発明においては、請求項20の発明において、まず始めに基準位置の多重化ヘッダ抽出処理を行い、次に基準位置の前後 J ビットシフトした範囲で多重化ヘッダの抽出処理を行い、以下順次基準位置の前後 P ビット（ P は整数）シフトした範囲まで J ビットきざみで多重化ヘッダの抽出処理を行うこととしている。

【0039】請求項22記載の発明においては、請求項21の発明において、多重化ヘッダの抽出処理の結果、ヘッダ情報に誤りのない多重化ヘッダが抽出され次第、その基準位置での以下の多重化ヘッダ抽出処理は中断することとしている。

【0040】請求項23記載の発明においては、請求項20から請求項22の発明のいずれかにおいて、抽出されたヘッダ以下の $F \times N$ バイトを情報フィールドとみなして多重分離を行うこととしている。

【0041】請求項24記載の発明においては、請求項20から請求項23記載の発明のいずれかにおいて、抽出された多重化ヘッダの位置が基準位置から p ビット（ p は整数）シフトする状態が S 回（ S は自然数）連続した場合、次の多重化フレームの基準位置を p ビットシフトすることとしている。

【0042】請求項25記載の発明においては、請求項20から請求項24の発明のいずれかにおいて、基準位

置の前後Pビットの範囲から多重化ヘッダが抽出されない状態がE1回(E1は自然数)連続した場合に同期外れとみなして同期フラグを有するF×Nバイト(F, Nは自然数)の同期フレームを用いてフレーム同期を確立する手順を実行することとしている。

【0043】請求項26記載の発明においては、請求項20から請求項25の発明のいずれかにおいて、正常時にはP=0として基準位置のみの多重化ヘッダ抽出処理のみを行っており、その結果、多重化ヘッダが抽出されない状態がE2回(E2は自然数)連続した場合にP≥1に設定し、その結果、抽出された多重化ヘッダの位置が基準位置からpビット(pは整数)シフトする状態がS回(Sは自然数)連続した場合、次の多重化フレームの基準位置をpビットシフトし、P=0とした正常状態に移行することとしている。

【0044】請求項27記載の発明においては、同期フラグを有するF×Nバイト(F, Nは自然数)の多重化フレームから同期フラグを有しないF×Nバイトの多重化フレームに移行する場合に、同期フラグを有する多重化フレームの受信と並行して、多重化ヘッダ抽出処理をF×Nバイト毎に行うこととしている。

【0045】請求項28記載の発明においては、同期フラグを有するF×Nバイト(F, Nは自然数)の多重化フレームから同期フラグを有しないF×Mバイト(MはN以外の自然数)の多重化フレームに移行する場合に、同期フラグを有する多重化フレームの受信と並行して、多重化ヘッダ抽出処理をF×Nバイト毎に行い、Fバイト毎またはF×Lバイト毎(LはNとMの最大公約数)に多重化ヘッダ抽出処理を行うこととしている。

【0046】請求項29記載の発明においては、請求項27または請求項28の発明において送信側から同期フラグ変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側は確認応答受信後に同期フラグを変更する場合に、受信側は確認応答返送後、多重化ヘッダ抽出処理をF×NバイトおきまたはFバイトおきまたはF×Lバイトおきに実行することとしている。

【0047】請求項30記載の発明においては、請求項27から請求項29の発明のいずれかにおいて送信側から同期フラグ変更コマンドを送信し、受信側がこれに対して確認応答を返送し、送信側はあらかじめ応答に要する応答遅延時間を測定しており、送信側は応答遅延時間の測定結果を受信側に通知しており、送信側は確認応答受信後に同期フラグを変更する場合に、受信側は確認応答返送時から応答遅延時間経過後、多重化ヘッダ抽出処理をF×NバイトおきまたはFバイトおきまたはF×Lバイトおきに実行することとしている。

【0048】請求項31記載の発明においては、請求項27から請求項30の発明のいずれかにおいて、多重化フレームの先頭に多重化ヘッダがS回連続して抽出されるまでは、同期フラグを有する多重化フレームの受信を

並行して行うこととしている。

【0049】請求項32記載の発明においては、請求項27から請求項31の発明のいずれかにおいて、多重化ヘッダの誤り訂正能力または誤り検出能力またはその双方を同期フラグの変更と同時に変更する場合において、多重化フレームの受信は変更前の誤り訂正能力および誤り検出能力で多重化ヘッダの誤り訂正および誤り検出を行い、多重化フレームの先頭にて行う多重化ヘッダ抽出処理は変更後の誤り訂正能力および誤り検出能力で多重化ヘッダの抽出処理を行うこととしている。

【0050】請求項33記載の発明においては、請求項31または請求項32記載の発明において、同期フラグおよび多重化ヘッダ以外のパラメータの変更は多重化フレームの先頭に多重化ヘッダがS回連続して抽出された時に降に行うこととしている。

【0051】

【発明の実施の形態】以下、本発明の実施の形態について、図面を参照しながら説明を行う。

【0052】(実施の形態1) 図1は先頭に同期フラグを有する場合の多重化フレームのフレーム長を80バイトから60バイトに変更する時の本発明の実施の形態1に係わるタイミング図である。PHSの無線スロットは20バイトであり、F=20とすると都合がよい。図1において、(a)は多重化フレーム長を60バイトに変更する時の1フレーム後に変更された場合のタイミング図である。(b)は多重化フレーム長を60バイトに変更する時の2フレーム後に変更された場合のタイミング図である。(c)は多重化フレーム長を60バイトに変更する時の3フレーム後に変更された場合のタイミング図である。(d)は多重化フレーム長を60バイトに変更する時の4フレーム後に変更された場合のタイミング図である。(e)は本発明の実施の形態1に係る多重化フレーム長変更時の同期フラグ抽出処理のタイミング図である。図2は先頭に同期フラグを有する場合の多重化フレームのフレーム長を80バイトから40バイトに変更する時の本発明の実施の形態1に係わるタイミング図である。図2において、(a)は多重化フレーム長を40バイトに変更する時の1フレーム後に変更された場合のタイミング図である。(b)は多重化フレーム長を40バイトに変更する時の2フレーム後に変更された場合のタイミング図である。(c)は多重化フレーム長を40バイトに変更する時の3フレーム後に変更された場合のタイミング図である。(d)は多重化フレーム長を40バイトに変更する時の4フレーム後に変更された場合のタイミング図である。(e)は本発明の実施の形態1に係る多重化フレーム長変更時の同期フラグ抽出処理のタイミング図である。図1において多重化フレームは従来の技術で図9を用いて説明しているH.223/Annex Aで規定されている多重化フレームであり、ここではその説明を省略する。同期と書かれた部分は同期フラグであ

り、その他の部分にはヘッダおよび情報フィールドが格納されている。以下これらのタイミング図および図3に示すフローチャートを用いてその動作を説明する。

【0053】図1においてハッチングをほどこした同期フラグは60バイトのフレーム長の多重化フレームの同期フラグである。図1(a)から図1(d)まではフレーム長が80バイトから60バイトに切り替わるタイミングが1フレームずつづれている場合を示している。ここで、ハッチングをほどこした同期フラグに着目すると(a)に対して(b)の同期フラグは20バイトおくれであり、(b)に対して(c)の同期フラグは同様に20バイトおくられている。さらに、(c)に対して(d)の同期フラグはさらに20バイトおくられているが、これは(a)の同期フラグと同じタイミングである。つまり60バイトの同期フラグは20バイトずれた3つの位相のいずれかにあり、それ以前の80バイトの同期フラグはすべて(a)から(c)の60バイトの同期フラグの中のいずれかと同じタイミングであることがわかる。このことから、80バイトから60バイトに多重化フレームのフレーム長を変更する場合には、20バイトごとに同期フラグ抽出処理を行うことが最低限必要であることがわかる。同期フラグ抽出処理とは、受信データを1ビットずつシフトしながら予め決められたビット数の同期パターンと受信データを比較してパターンマッチングを行うものであり、同期パターンとしては、初期状態では4バイトの同期フラグ「1111100011011101010000100101100」を用いることになっている。

【0054】H. 223/AnnexAでは多重化フレームのフレーム長を半固定とすることにより、同期フラグ抽出処理を固定長のフレーム長毎にその前後数ビットに対してのみ行うようにすることで処理の軽減と伝送誤りによる誤抽出の低減を実現している。数ビットのサーチは交換機のクロック精度の誤差によりおこる網の同期スリップを吸収する範囲内に選ばれる。

【0055】次に図2に示すように80バイトのフレーム長を40バイトに変更する場合を考える。この場合にはハッチングをほどこした40バイトの同期フラグは40バイト毎に整列しており、それ以前の80バイトの同期フラグは80バイト毎に整列している。つまり、40バイト毎に同期フラグ抽出処理を行えばいいことになる。もちろん、この場合にも20バイト毎に同期フラグ抽出処理を行っていったら同様に同期の抽出は行えるが、40バイト毎に処理を行うほうが処理が少なくてすむ。以上の考察から、 $20 \times N$ バイト(N は自然数)の多重化フレームから $20 \times M$ バイト(M は N 以外の自然数)の多重化フレームにフレーム長を変更する場合には、20バイト毎に同期フラグ抽出処理を行うか、または N と M の最大公約数を L とした場合に $20 \times L$ バイト毎に同期フラグ抽出処理を行えばよい。さらにこれらの同期フ

ラグ抽出処理は、従来の技術図15で説明したH. 223/AnnexAで規定されている多重化フレームのフレーム長を変更するコマンドおよびシーケンスにおいては、受信側においてフレーム長変更コマンドに対する確認応答返送後から行えばよく、フレーム長変更後の同期フラグが規定された保護段数だけ抽出された時点で変更後のフレーム長の固定長フレームの同期フラグ抽出処理に移行すればよい。以上の動作を記述したフローチャートを図3に示す。図3において301は多重化フレーム長変更前の状態を示しており、 $20 \times N$ バイト周期で同期フラグ抽出処理を行っている。カッコ内の数字は一例を示している。この状態はフレーム長変更確認応答を返送するまで繰り返される。302の判断でYesの場合、 $20 \times L$ バイトで同期フラグ抽出処理を行うモードに移行する。303は同期保護のための変数 s を初期化している。304は $20 \times M$ バイトの同期フラグ間隔を抽出する変数 m を初期化している。305は $20 \times L$ バイト後の同期フラグ抽出処理を行っている。306の判断の結果、同期フラグがない場合には307で m がインクリメントされる。306の判断の結果、同期フラグが抽出された場合、308で一つ前の同期フラグと今回抽出された同期フラグにはさまれた領域をヘッダおよび情報フィールドとして抽出し上位層へ転送する。309は同期フラグの間隔が $20 \times M$ バイトであったかどうか判断している。309の判断の結果、同期フラグの間隔が $20 \times M$ バイトでなかったら311で同期保護のための変数 s を初期化して304で同期フラグ間隔を抽出する変数 m を初期化して $20 \times L$ バイトの同期フラグ抽出処理305を続行する。309の判断の結果、同期フラグの間隔が $20 \times M$ バイトであったなら310において同期保護のチェックを行う。 S はあらかじめ設定されている同期保護のための連続同期フラグ抽出数であり、 s が S に満たない場合には312で s がインクリメントされ、304で同期フラグ間隔を抽出する変数 m を初期化して $20 \times L$ バイトの同期フラグ抽出処理305を続行する。310で $s = S$ の場合は変更後のフレーム長の同期が確立したものとみなし、 $20 \times M$ バイト周期の受信313に移行する。

【0056】以上のように本実施の形態では、同期フラグを有する半固定長の多重化フレームのフレーム長を変更する場合において少ない処理量で速やかにフレーム長の変更を行うことが可能となる。

【0057】なお、MODE ARQのように確認応答を返送してからその結果として変更後のフレーム長が返送されるまでの応答遅延時間を事前に測定してわかっている場合には、応答遅延時間返送時点から応答遅延時間の後に20バイト毎または $20 \times L$ バイト毎の同期フラグ抽出処理に移行するようにすればより処理量は軽減される。また、本実施の形態では $F = 20$ の場合について説明したが、 F の値がその他の場合においても、さらに、 N や M

の値がここであげた値以外の場合でも本発明は同様に効果を発揮する。

【0058】なお、ここでは多重化ヘッダ抽出処理のオフセットを $J=1$ ビットきざみでシフトする場合を例にあげて説明したが、網の同期ずれが4ビット毎に起こる場合には $J=4$ として基準位置の前後を4ビットきざみで多重化ヘッダの抽出処理を行うようにすればよい。

【0059】(実施の形態2) 図4は先頭に同期フラグを有さない場合の多重化フレームのフレーム長を80バイトから60バイトに変更する時の本発明の実施の形態2に係わるタイミング図である。PHSの無線スロットは20バイトであり、 $F=20$ とすると都合がよい。図4において、(a)は多重化フレーム長を60バイトに変更する時の1フレーム後に変更された場合のタイミング図である。(b)は多重化フレーム長を60バイトに変更する時の2フレーム後に変更された場合のタイミング図である。(c)は多重化フレーム長を60バイトに変更する時の3フレーム後に変更された場合のタイミング図である。(d)は多重化フレーム長を60バイトに変更する時の4フレーム後に変更された場合のタイミング図である。(e)は本発明の実施の形態2に係る多重化フレーム長変更時の多重化ヘッダ抽出処理のタイミング図である。図4において多重化フレームは従来の技術で図12を用いて説明しているH.223/Annex Aを拡張して同期フラグを削除した多重化フレームであり、ここではその説明を省略する。MCと書かれた部分は従来の技術図10で示した多重化ヘッダであり、その他の部分には情報フィールドが格納されている。以下タイミング図4および図5に示すフローチャートを用いてその動作を説明する。

【0060】図4においてハッチングをほどこした多重化ヘッダは60バイトのフレーム長の多重化フレームの多重化ヘッダである。図1(a)から図1(d)まではフレーム長が80バイトから60バイトに切り替わるタイミングが1フレームずつづれている場合を示している。ここで、ハッチングをほどこした多重化ヘッダに着目すると(a)に対して(b)の多重化ヘッダは20バイトおぐれており、(b)に対して(c)の多重化ヘッダは同様に20バイトおぐれている。さらに、(c)に対して(d)の多重化ヘッダはさらに20バイトおぐれているが、これは(a)の多重化ヘッダと同じタイミングである。つまり60バイトの多重化ヘッダは20バイトずれた3つの位相のいずれかにあり、それ以前の80バイトの多重化ヘッダはすべて(a)から(c)の60バイトの多重化ヘッダの中のいずれかと同じタイミングであることがわかる。このことから、80バイトから60バイトに多重化フレームのフレーム長を変更する場合には、20バイトごと多重化ヘッダ抽出処理を行うことが最低限必要であることがわかる。多重化ヘッダ抽出処理とは、多重化ヘッダに添付された誤り訂正符号を用

いて多重化情報部と誤り検出符号の誤り訂正を行った後、誤り検出符号を用いて多重化情報部の誤り検出し、誤りが検出されない場合に、多重化ヘッダとして抽出するようにする。さらに多重化ヘッダ抽出処理の中で交換機のクロック精度の誤差によりおこる網のビットスリップを吸収する範囲内で数ビットのサーチを行ってもかまわないし、おこなわなくてもよい。

【0061】図4の結果は実施の形態1の図1の結果と類似しており、同様に80バイトから40バイトにフレーム長を変更する場合には図2と同じ結果となるため図による説明は省略するが、 $20 \times N$ バイト(N は自然数)の多重化フレームから $20 \times M$ バイト(M は N 以外の自然数)の多重化フレームにフレーム長を変更する場合には、20バイト毎に多重化ヘッダ抽出処理を行うか、または N と M の最大公約数を L とした場合に $20 \times L$ バイト毎に多重化ヘッダ抽出処理を行えばよい。さらにこれらの多重化ヘッダ抽出処理は、従来の技術図15で説明したH.223/Annex Aで規定されている多重化フレームのフレーム長を変更するコマンドおよびシーケンスにおいては、受信側においてフレーム長変更コマンドに対する確認応答返送後から行えばよく、フレーム長変更後の多重化ヘッダが規定された保護段数だけ抽出された時点で変更後のフレーム長の固定長フレームの多重化ヘッダ抽出処理に移行すればよい。以上の動作を記述したフローチャートを図5に示す。図5において501は多重化フレーム長変更前の状態を示しており、 $20 \times N$ バイト周期で多重化ヘッダ抽出処理を行っている。カッコ内の数字は一例を示している。この状態はフレーム長変更確認応答を返送するまで繰り返される。502の判断でYesの場合、 $20 \times L$ バイトで多重化ヘッダ抽出処理を行うモードに移行する。503は同期保護のための変数 s を初期化している。504は $20 \times M$ バイトの多重化ヘッダ間隔を抽出する変数 m を初期化している。505は $20 \times L$ バイト後の多重化ヘッダ抽出処理を行っている。506の判断の結果、多重化ヘッダがない場合には507で m がインクリメントされる。506の判断の結果、多重化ヘッダが抽出された場合、508で一つ前の多重化ヘッダと今回抽出された多重化ヘッダにはさまれた領域を情報フィールドとして抽出し上位層へ転送する。509は多重化ヘッダの間隔が $20 \times M$ バイトであったかどうか判断している。509の判断の結果、多重化ヘッダの間隔が $20 \times M$ バイトでなかったら511で同期保護のための変数 s を初期化して504で多重化ヘッダ間隔を抽出する変数 m を初期化して $20 \times L$ バイトの多重化ヘッダ抽出処理505を続行する。509の判断の結果、多重化ヘッダの間隔が $20 \times M$ バイトであったなら510において同期保護のチェックを行う。 S はあらかじめ設定されている同期保護のための連続多重化ヘッダ抽出数であり、 s が S に満たない場合には512で s がインクリメントされ504で多重化ヘッダ間隔

を抽出する変数 m を初期化して $20 \times L$ バイトの多重化ヘッダ抽出処理505を続行する。510で $s = S$ の場合は変更後のフレーム長の同期が確立したものとみなし、 $20 \times M$ バイト周期の受信513に移行する。

【0062】以上のように本実施の形態では、同期フラグを有さない場合においても半固定長の多重化フレームのフレーム長を少ない処理量で速やかに変更することが可能となる。

【0063】なお、MODS ARQのように確認応答を返送してからその結果として変更後のフレーム長が返送されるまでの応答遅延時間を事前に測定してわかっている場合には、応答遅延時間返送時点から応答遅延時間の後に20バイト毎または $20 \times L$ バイト毎の同期フラグ抽出処理に移行するようにすればより処理量は軽減される。また、本実施の形態では $F = 20$ の場合について説明したが、 F の値がその他の場合においても、さらに、 N や M の値がここで上げた値以外の場合でも本発明は同様に効果を発揮する。

【0064】また、H. 223AnnexAでは多重化ヘッダの誤り耐性を強化する目的で多重化ヘッダを多重化フレーム内にインタリーブするモードがあるが、フレーム長が不明の場合、インタリーブを戻せなくなるため、本発明を実施する場合には前もって多重化ヘッダをインタリーブしないモードに設定しておく必要がある。

【0065】(実施の形態3) 図6は、本発明の実施の形態3に係る多重化ヘッダ抽出方法の動作原理を示すタイミング図である。

【0066】図6は、同期フラグを有する $20 \times N$ バイト(N は自然数)の同期フレームを用いてフレーム同期を確立した後、多重化フレームのフレーム長を半固定にし、デジタル網のクロック同期を利用することにより同期フラグを有しない多重化フレームにより音声データまたはビデオデータまたはその他のデータまたはそれらの任意の組み合わせを多重化して伝送する場合の多重化ヘッダ抽出方法の動作原理を説明している。図7は、本発明の実施の形態3に係る多重化ヘッダ抽出方法の動作を示すフローチャートである。

【0067】図8は、本発明の実施の形態3に係る多重化ヘッダ抽出方法の動作を示すフローチャートである。

【0068】以下、図6に示すタイミング図および図7、図8に示すフローチャートを用いてその動作を説明する。

【0069】図6(a)は網が正常な場合の多重化ヘッダ抽出方法を示しており、一つ前に抽出された多重化ヘッダを基準に $20 \times N \times 8$ ビット後にヘッダがあるとしてヘッダを抽出すれば1回目でも多重化ヘッダを抽出できる。多重化ヘッダか否かの判断は、例えばH. 223/AnnexAの場合、図10に示す誤り訂正符号部の訂正符号を用いて多重化情報部および誤り検出符号部の誤り訂正を行った後に誤り検出符号部による多重化情報部の誤り検出

を行い、その結果、誤りが検出されなければそれを多重化ヘッダとみなし、以下に続く $20 \times N$ バイトから多重化ヘッダ部のバイト数を引いた残りのバイト数を情報フィールドとして切り出し、多重化情報部に従った多重分離を行う。図6(b)は一つ前の多重化ヘッダを抽出した後、網でビット抜けが発生した場合である。ビット抜けは交換機のクロック精度の誤差により発生する。この場合、一つ前に抽出された多重化ヘッダから $20 \times N \times 8$ ビット後のヘッダ基準位置にはヘッダはなく、それよりビット抜けしたビット数分だけ前方に位置する。これとは逆にビット挿入が発生した場合には多重化ヘッダはヘッダ基準位置より後方に位置する。ヘッダの存在確立はヘッダ基準位置が最も高く、それから遠ざかるほど低くなる。また、多重化ヘッダの抽出処理は上述したように複雑であるためできるだけその回数を減らす必要がある。そこで、一般的な同期フラグの抽出処理のように受信した順にビットをシフトしてパターンマッチングを行う場合とは異なり、ヘッダ基準位置から始めて、 ± 1 ビット、 ± 2 ビット... $\pm P$ ビットまでといった順に多重化ヘッダ抽出処理を行うようにする。ここで P は整数であり、網のビットずれの発生状況に応じて予めまたは適応的に決定される。また $P = 0$ はヘッダの抽出処理をヘッダ基準位置でしか行わないことを意味する。こうすることにより正しいヘッダの存在確立の高い順にサーチしていることになり、誤りのない多重化ヘッダを抽出し次第、多重化ヘッダ抽出処理を中断することができ、ビットずれがない場合には通常と同じ1回の多重化ヘッダ抽出処理で済み、処理量を大幅に低減できる。また、ビットずれがおこった後はしばらく $20 \times N$ バイトの間隔で多重化ヘッダが到着するため、ヘッダは常にビットずれが発生する前のヘッダ基準位置とはずれたままとなる。これでは、ビットずれが発生しなくてもヘッダ抽出処理を複数回実行する必要がある、また同じ方向にビットずれを複数回起こした場合には $\pm P$ の窓から外れてしまうことになる。そこで、多重化フラグが抽出された位置とヘッダ基準位置との位相関係が S 回同じ状態が続けばヘッダ基準位置を多重化フラグが抽出された位置とすることにより上述した課題が解決できる。ここで S は自然数であり、網のビットずれの頻度やビットエラーの発生度合いに応じて事前にまたは適応的に決定される。

【0070】次に以上の動作を図7のフローチャートを用いて説明する。図7において701、702では変数の初期値を設定している。703、704は基準位置から p のオフセット位置で多重化ヘッダ抽出処理を行う。705は多重化ヘッダか否かの判断を行っており、多重化ヘッダの抽出できなかった場合には706においてオフセット値 p がオフセット窓の最後か否かを判断し、途中の場合には707において次のオフセット値となる p の値をセットする。707ではひとつ前の p の値が0以下の場合、上段の計算式を、前の p の値が正の整数の場合

合下段の計算式を使用する。706で $p = P$ になった場合、基準より設定したオフセット値の範囲内ではヘッダが抽出されなかった場合を示している。これは、極端なビットずれによりサーチ窓の範囲から逸脱した場合と、多重化ヘッダに誤り訂正能力以上の伝送誤りが発生した場合に起こる。

【0071】709から714は多重化フラグが抽出された位置とヘッダ基準位置との位相関係が S 回同じ状態が続けばヘッダ基準位置を多重化フラグが抽出された位置とする処理を行っている。709は抽出されたオフセット値が前回と同じか否かを判断している。同じ場合には710でカウンタ s をインクリメントし、711では s が規定回数 S に到達したか否かを判断する。712はオフセット値が連続しない場合にカウンタ s をクリアする。714では今のオフセット値を p に待避したのち、オフセット値 p をクリアする。715では次の基準位置を算出しており、711の判断で基準位置をずらすことに決定した場合のみその時の p の値が加算される。708は誤りが検出されなかった多重化ヘッダを基準に以下に続く $20 \times N$ バイトから多重化ヘッダ部のバイト数を引いた残りのバイト数を情報フィールドとして切り出しており、多重化処理としては多重化情報部に従った多重分離を行う。

【0072】次に極端なビットずれによりサーチ窓の範囲から逸脱した場合、これ以上前述した処理を続けていても多重化ヘッダは抽出できない。そこで、基準位置の前後 P ビットの範囲から多重化ヘッダが抽出されない状態が $E1$ 回($E1$ は自然数)連続した場合に同期外れとみなして同期フラグを有する $20 \times N$ バイト(N は自然数)の同期フレームを用いてフレーム同期を確立する手順を実行する。また、多重化ヘッダに誤り訂正能力以上の伝送誤りが発生した場合には、多重化ヘッダは基準位置にあっても抽出されない。しかし、多くの処理を費やしてその周辺をサーチしてもやはり多重化ヘッダは抽出されないのは明白である。そこで、正常時には $P = 0$ として基準位置の多重化ヘッダ抽出処理のみを行っており、その結果、多重化ヘッダが抽出されない状態が $E2$ 回($E2$ は自然数)連続した場合に $P \geq 1$ に設定し、その結果、抽出された多重化ヘッダの位置が基準位置から p ビット(p は整数)シフトする状態が S 回(S は自然数)連続した場合、次の多重化フレームの基準位置を p ビットシフトし、 $P = 0$ とした正常状態に移行することになれば、伝送エラーに左右されることがなくより少ない処理量で多重化ヘッダの抽出処理が可能となる。

【0073】その動作を図8のフローチャートを用いて説明する。図8は816の判断以降の処理を除いては図7と同じであり説明を省略する。801で最初、サーチ窓のサイズ $P = 0$ であり、基準位置で多重化ヘッダが抽出されない場合はすべて、816に導かれる。816はサーチ窓のサイズが0か $P1$ かを判断している。 $P1$ は

事前にまたは適応的に設定された自然数である。 $e2$ はこの状態で何回連続して多重化ヘッダの抽出ができないかをカウントするカウンタであり、818でインクリメントされる。820は連続して多重化ヘッダを抽出できない回数が予めまたは適応的に設定された $E2$ に到達したか否かを判断している。 $e2$ が $E2$ に達すると823で $P = P1$ に設定され、多重化ヘッダ抽出処理のサーチ窓が拡大される。 $e1$ はサーチ窓拡大後に連続して多重化ヘッダが抽出できなかった数をカウントするカウンタであり、817でインクリメントされる。819はサーチ窓拡大後に連続して多重化ヘッダが抽出できなかった数が予めまたは適応的に設定された $E1$ に達したか否かを判断している。819の判断の結果、サーチ窓拡大後に連続して多重化ヘッダが抽出できなかった数が予めまたは適応的に設定された $E1$ に達した場合、同期はずれ812としてこの処理を終え、同期フラグを有する $20 \times N$ バイト(N は自然数)の同期フレームを用いてフレーム同期を確立する手順を実行する。

【0074】以上のように、本実施の形態では、伝送エラーが発生する場合においても安定して、少ない処理量で多重化ヘッダの抽出を行うことが可能となり、同期フラグを付けない高効率な伝送を享受しつつ、再同期の発生頻度を大幅に削減できる。

【0075】(実施の形態4)図17は先頭に同期フラグを有する多重化フレームを先頭に同期フラグを持たない多重化フレームに変更する時の本発明の実施の形態4に係わるタイミング図である。PHSの無線スロットは20バイトであり、 $F = 20$ とすると都合がよい。図17において、(a)は同期フラグの変更要求に対して受信側が返送した確認応答が1度で送信側に届いた場合のタイミング図である。(b)は同期フラグの変更要求に対して受信側が返送した確認応答が誤り、送信側で確認応答待ちタイムアウトの結果再送した同期フラグの変更要求に対する2度目の確認応答が送信側に届いた場合のタイミング図である。以下これらのタイミング図および図18に示すフローチャートを用いてその動作を説明する。

【0076】図17において同期と書かれているのは多重化フレームの同期フラグであり、ここでは2バイトであることを示す。またMCと書かれているのは多重化コードであり、誤り検出コード、誤り訂正コードで伝送誤りに対して保護されており、やはり2バイトであるとする。同期フラグ長や誤り検出能力を決めるコード長、誤り訂正能力を決めるコード長は選択可能である。図1

(a)で1フレーム目は同期フラグのついたフレームを受信している場合であり、同期フラグを抽出してそれに続く多重化ヘッダを抽出し、以下に続く情報フィールドを切り出す。同期フラグの抽出は前後の数ビットをシフトしながらパターンマッチングをしても良いし、先頭の例えば2バイトを切り出してパターンマッチングをする

だけでも良い。図1の1フレーム目の情報フィールドには同期フラグを2バイトから同期フラグなしに変更する要求が格納されている。これを受信した受信側はその可否を判断し、即座に確認応答を返送する。図1では確認応答返送後2フレーム後にあたる3フレーム目に変更された多重化フレームが届いているが受信側ではこのタイミングはわからない。よって、確認応答返送後、すぐに同期フラグがあるフレームの受信に並行して先頭に多重化フレームが格納された同期フラグのないフレームの受信を行う。2フレーム目は同期フラグのある多重化フレームのため、この場合、同期フラグのあるフレームの受信で多重化ヘッダおよび情報フィールドが切り出される。3フレーム目は多重化フレームが同期フラグのないフレームに変更されているため、同期フラグのあるフレームの受信側は多重化フレームおよび情報フィールドの抽出ができない。一方、多重化フレームで始まるフレームの受信側は多重化ヘッダが抽出される。ここでは $S=2$ 回連続して多重化ヘッダで始まるフレームが受信側で抽出された場合、以降の受信を多重化ヘッダで始まるフレームの受信のみを行うことにしているため、5フレーム目の受信は多重化ヘッダで始まるフレームの受信のみを行う。図1(b)は2フレームまでの動作は同じであるが、受信側が1回目に返送した確認応答が伝送誤りのために送信側に届かなかったため、3フレーム目は同期フラグのあるフレームである。この場合、受信側では同期フラグのあるフレームの受信で多重化ヘッダおよび情報フィールドの切り出しができる。その後、送信側で最初に変更要求を送信した時にセットした確認応答待ちのタイマが作動するため、送信側は4フレーム目に変更要求を再度送信している。この場合にも受信側においては同期フラグのあるフレームの受信で多重化ヘッダおよび情報フィールドの切り出しができるため、変更要求の再送信を受信することができる。図1(b)において変更要求の再送信に対する2度目の確認応答は送信側に届いており、5フレーム以降は図1(a)の2フレーム目以降と同様の動作をする。

【0077】以上の動作を記述したフローチャートを図18に示す。図18において1801は同期フラグ変更前の状態を示しており、 $20 \times N$ バイト周期で同期フラグ抽出処理を行っている。カッコ内の数字は一例を示している。この状態は受信側が同期フラグ変更確認応答を返送するまで繰り返される。1802の判断でYesの場合、 $20 \times N$ バイトで先頭の2バイトを抽出する処理を行うモードに移行する。1803は同期保護のための変数 s を初期化している。1804は $20 \times N$ バイト後の先頭の2バイトの抽出処理を行っている。1805の判断の結果、2バイトが同期フラグでない場合には1807へ移行する。1805の判断の結果、同期フラグが抽出された場合、それに続く領域をヘッダおよび情報フィールドとして抽出し上位層へ転送する。上位層の解析

の結果、受信した情報フィールドが同期フラグ変更要求であった場合、確認応答を返送する。1807は先頭の2バイトから多重化ヘッダを抽出する。多重化ヘッダの抽出は添付された誤り訂正コードで誤り訂正を行った後、添付された誤り検出コードによる誤り検出の結果、誤りが検出されなければ次の1808においてこの2バイトを多重化ヘッダとみなす。1808の判断の結果、先頭の2バイトが多重化ヘッダでない場合、1809で s がリセットされ次のフレームの受信に移る。1808の判断の結果、先頭の2バイトが多重化ヘッダの場合、1810において多重化ヘッダ抽出の連続回数が所定の値 S に到達したか否かを判断する。 $s=S$ の場合、1812に移行して以後、同期フラグなしの受信を続ける。 $s < S$ の場合、1811で s がインクリメントされ、次のフレームの受信に移る。

【0078】以上のように本実施の形態では、同期フラグを有する多重化フレームから同期フラグをもたない多重化フレームに変更する場合において確実にかつ速やかに同期フラグの変更を行うことが可能となる。

【0079】なお、MODS ARQのように確認応答を返送してからその結果として変更後のフレーム長が返送されるまでの応答遅延時間が事前に測定してわかっている場合には、応答遅延時間返送時点から応答遅延時間の後に先頭の2バイトが多重化ヘッダか判断する処理に移行するようにすればより処理量は軽減される。また、本実施の形態では $F=20$ の場合について説明したが、 F の値がその他の場合においても、さらに、 N や M の値がここであげた値以外の場合でも本発明は同様に効果を発揮する。

【0080】また、ここでは同期フラグも多重化ヘッダも2バイトの場合を例にあげて説明したが、2バイト以外でもかまわないし、それぞれが異なるバイト数でもかまわない。その場合、1804はそれぞれ個別に抽出し、同期フラグは1805で判断し、多重化ヘッダは1807、1808で判断すれば良い。また、ここでは同期フラグがある場合からない場合へ変更する場合を例に説明しているが、同期フラグがない場合から同期フラグがある場合に変更することも可能である。この場合、1801で同期フラグのないフレームの受信を行い、1812で同期フラグのあるフレームの受信を行うようにすれば良い。

【0081】また、同期フラグの変更と同時に多重化フレーム長を $F \times M$ バイトに変更する場合にも、本実施の形態に加えて F バイト毎、または $F \times L$ バイト(L は N と M の最大公約数)毎に多重化ヘッダの抽出処理を行うことにより可能となる。

【0082】また、ここでは処理が簡単になるように同期フラグのあるフレームの受信処理の途中から同期フラグのないフレームの受信処理に分岐させているが、両方の処理をおこなった後に総合的に判断することにしても

よい。例えば、1805において同期フラグのある多重化フレームの先頭に同期フラグが抽出された場合にも抽出されない場合にもそれに続く1806でヘッダおよび情報フィールド抽出処理を行った後に多重化ヘッダが抽出されない場合に1807に分岐することによれば同期フラグに伝送誤りが発生した場合にも多重化ヘッダの抽出が可能であり、より伝送誤りに強くすることができる。さらに1806で多重化ヘッダが抽出された場合でもそれに続く情報フィールドを上位層に転送し、1807に移行して同期フラグのない場合の多重化ヘッダの抽出処理を行い、こちらでも多重化ヘッダが抽出された場合には、どちらかの多重化ヘッダの誤検出であるが、特定はできないため変数sをインクリメントし、転送した情報フィールドは上位層の誤り検出機能にまかせることにしてもよい。

【0083】また、多重化ヘッダの誤り訂正能力、または誤り検出能力、またはその双方を同期フラグの変更と同時に変更する場合、送信側からのこれら変更のコマンドに確認応答を返送し、その直後、または所定の時間後に同期フラグのあるフレームの受信と同期フラグのないフレームの受信の双方を行なう際に、同期フラグのあるフレームの受信は変更前の誤り訂正能力、および誤り検出能力で1806のヘッダ抽出を行い、同期フラグのないフレームの受信は変更後の誤り訂正能力、および誤り検出能力で1807のヘッダ抽出を行なうようにすれば、同期フラグの変更と同時にできる。さらに、同期フラグの変更なしに多重化ヘッダの誤り訂正能力、または誤り検出能力、またはその双方を変更したい場合には、1805の判断をせず、変更前のパラメータと変更後のパラメータの双方で、所定の位置の多重化ヘッダの抽出処理を行い、S回連続して変更後のパラメータの多重化ヘッダが抽出されれば、変更後の受信のみに移行すればよい。

【0084】また、変更後の誤り訂正能力をなしにする場合も、これに含まれる。

【0085】

【発明の効果】上記の発明により、多重化フレーム長を変更した場合にも同期フラグの抽出処理が少なくなる。

【0086】また、同期フラグを有さない場合においても多重化フレームのフレーム長の変更が可能となる。

【0087】さらに、同期フラグを有する多重化フレームから同期フラグのない多重化フレームへの変更が確実に速やかに行える。

【0088】さらに、多重化ヘッダ抽出方法により同期フラグがない場合にも網のビットずれ発生に対して即座に同期を回復することが可能となる。

【図面の簡単な説明】

【図1】(a)は本発明の実施の形態1に係る多重化フレーム長を60バイトに変更する時の1フレーム後に変更された場合のタイミング図

10 (e)は本発明の実施の形態1に係る多重化フレーム長変更時の同期フラグ抽出処理のタイミング図

【図2】(a)は本発明の実施の形態1に係る多重化フレーム長を40バイトに変更する時の1フレーム後に変更された場合のタイミング図

(b)は本発明の実施の形態1に係る多重化フレーム長を40バイトに変更する時の2フレーム後に変更された場合のタイミング図

(c)は本発明の実施の形態1に係る多重化フレーム長を40バイトに変更する時の3フレーム後に変更された場合のタイミング図

20 (d)は本発明の実施の形態1に係る多重化フレーム長を40バイトに変更する時の4フレーム後に変更された場合のタイミング図

(e)は本発明の実施の形態1に係る多重化フレーム長変更時の同期フラグ抽出処理のタイミング図

【図3】本発明の実施の形態1に係る多重化フレーム長変更方法の動作を示すフローチャート

30 【図4】(a)は本発明の実施の形態2に係る多重化フレーム長を60バイトに変更する時の1フレーム後に変更された場合のタイミング図

(b)は本発明の実施の形態2に係る多重化フレーム長を60バイトに変更する時の2フレーム後に変更された場合のタイミング図

(c)は本発明の実施の形態2に係る多重化フレーム長を60バイトに変更する時の3フレーム後に変更された場合のタイミング図

(d)は本発明の実施の形態2に係る多重化フレーム長を60バイトに変更する時の4フレーム後に変更された場合のタイミング図

40 (e)は本発明の実施の形態2に係る多重化フレーム長変更時の同期フラグ抽出処理のタイミング図

【図5】本発明の実施の形態2に係る多重化フレーム長変更方法の動作を示すフローチャート

【図6】(a)は本発明の実施の形態3に係る多重化ヘッダ抽出方法の動作原理を示すタイミング図

(b)は本発明の実施の形態3に係るビットずれ発生時の多重化ヘッダ抽出方法の動作原理を示すタイミング図

【図7】本発明の実施の形態3に係る多重化ヘッダ抽出方法の動作を示すフローチャート

50 【図8】本発明の実施の形態3に係る多重化ヘッダ抽出

方法の動作を示すフローチャート

【図9】従来のH.223/AnnexAの多重化層の多重化フレームフォーマットを示す図

【図10】従来のH.223/AnnexAの多重化ヘッダのフォーマットを示す図

【図11】従来のMODS ARQのフレーム構成図

【図12】従来のH.223/AnnexAを拡張し、同期フラグを削除した多重化層の多重化フレームフォーマットを示す図

【図13】従来の80バイトの多重化フレームで音声、ビデオを多重した場合の多重化フレームの構成図

【図14】従来の60バイトの多重化フレームで音声、ビデオを多重した場合の多重化フレームの構成図

【図15】(a)は従来のフレーム長変更時の成功時のシーケンス図

(b)は従来のフレーム長変更時の伝送エラー時のシーケンス図

【図16】(a)は従来の多重化フレーム長を60バイトに変更する時の1フレーム後に変更された場合の同期*

*フラグ抽出のタイミング図

(b)は従来の多重化フレーム長を60バイトに変更する時の2フレーム後に変更された場合の同期フラグ抽出のタイミング図

(c)は従来の多重化フレーム長を60バイトに変更する時の3フレーム後に変更された場合の同期フラグ抽出のタイミング図

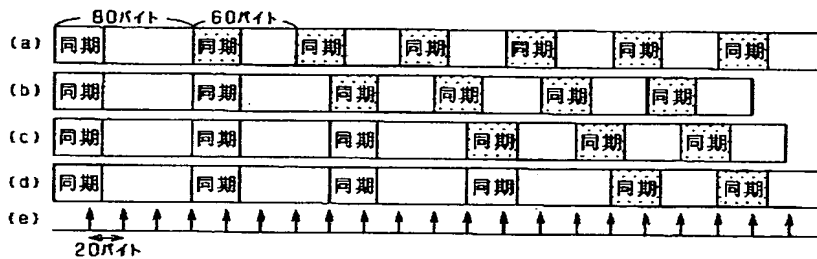
(d)は従来の多重化フレーム長を60バイトに変更する時の1フレーム後に変更された場合の多重化ヘッダ抽出のタイミング図

【図17】(a)は本発明の実施の形態4に係る同期フラグの変更方法のうち確認応答が送信側に届いた場合のタイミング図

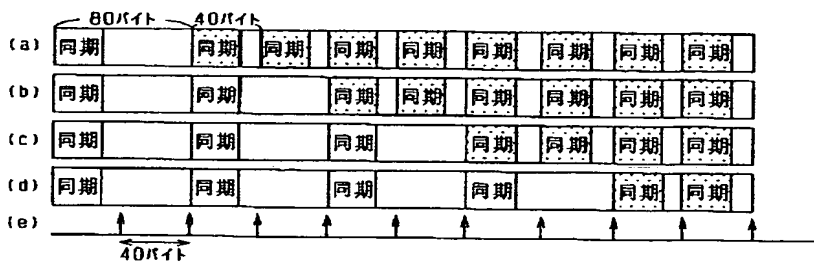
(b)は本発明の実施の形態4に係る同期フラグの変更方法のうち確認応答が送信側に届かない場合のタイミング図

【図18】本発明の実施の形態4に係る同期フラグの変更方法の動作を示すフローチャート

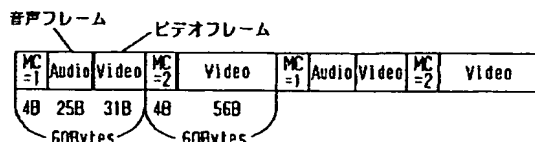
【図1】



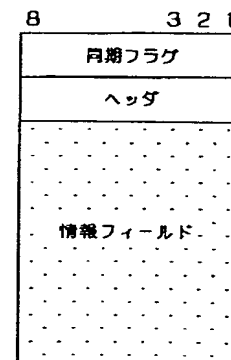
【図2】



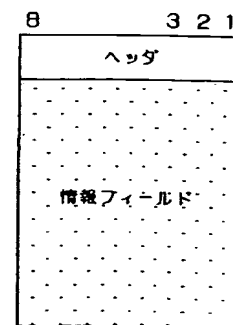
【図14】



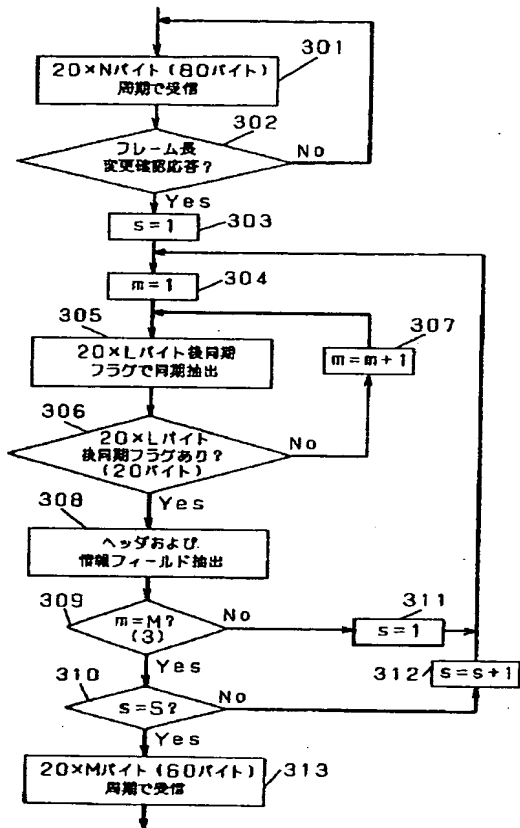
【図9】



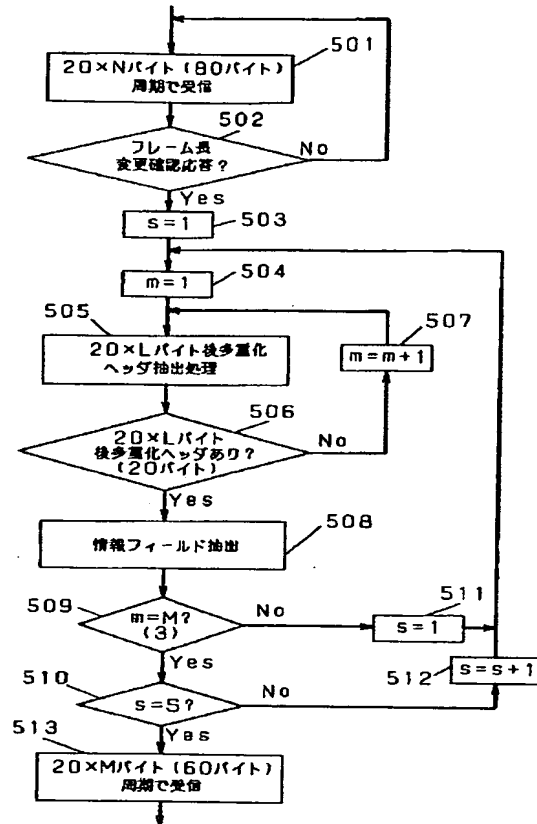
【図12】



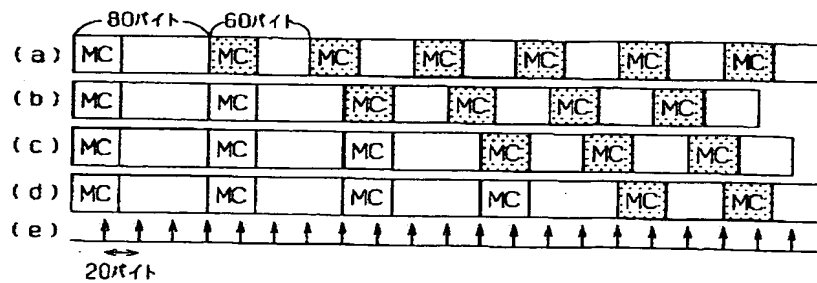
【図3】



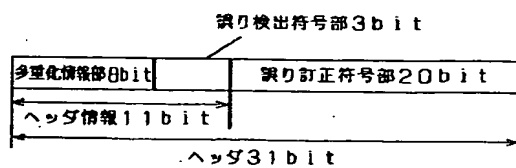
【図5】



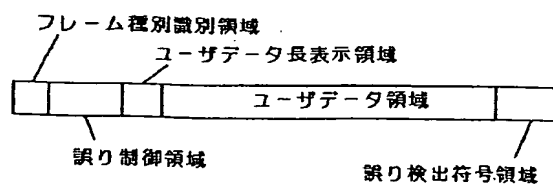
【図4】



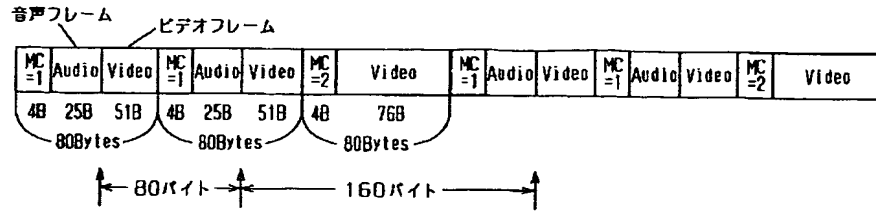
【図10】



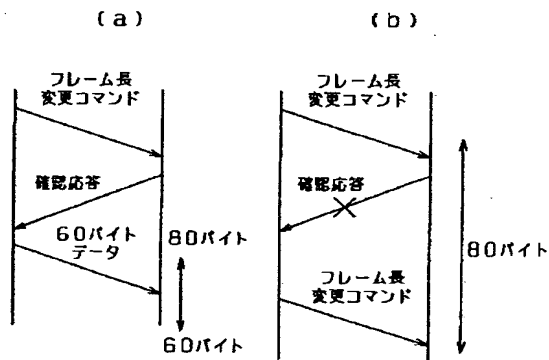
【図11】



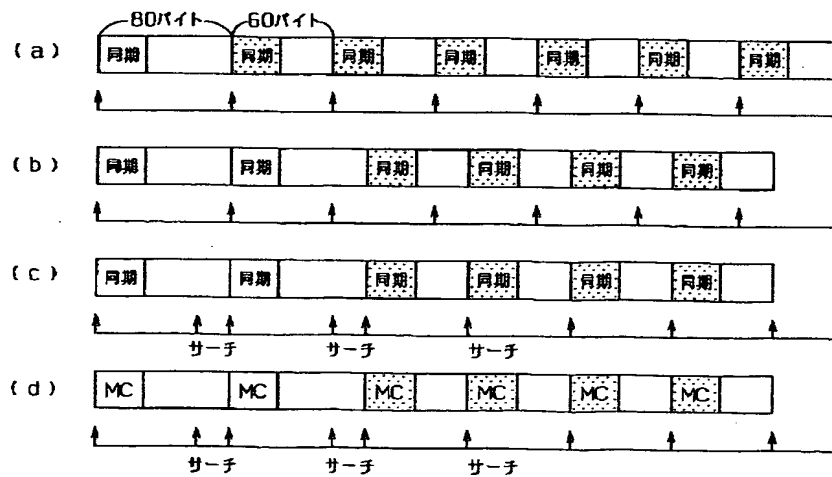
【図13】



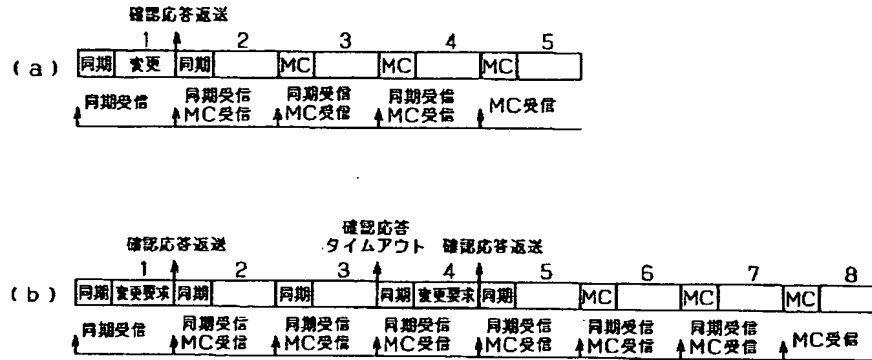
【図15】



【図16】



【図17】



【図18】

